

Further HCI

Alan Blackwell and Luke Church

Overview of the course

- **Theory driven approaches to HCI**
- Design of visual displays
- Goal-oriented interaction
- Designing efficient systems
- Designing smart systems
- Designing meaningful systems (guest lecturer)
- Evaluating interactive system designs
- Designing complex systems as interaction spaces

Lecture 1:

Theory driven approaches to HCI

What is a theory in HCI? Why take a theory driven approach to HCI?

Why theory in HCI?

HP Color LaserJet M452dn 19 x Luke

192.168.1.55/info_config_network.html?tab=Networking&menu...

HP Color LaserJet M452dn

HP Color LaserJet M452dn NPIC7F9E3 192.168.1.55

Home System Print Networking HP Web Services

Network Summary Configuration IPv4 Configuration IPv6 Configuration Network Identification Advanced Google Cloud Print Setup Proxy Settings AirPrint Status Security Settings Certificates HTTPS Enforcement SNMP Access Control List 802.1X Authentication Firewall

Network Summary

Shop for Supplies Support

TCP/IP(v4)

Status:	Ready
IPv4 Address:	192.168.1.55
Subnet Mask:	255.255.255.0
Default Gateway:	192.168.1.1
IP Configured By:	DHCP
IP Preferred Address Method:	DHCP
DHCP/BOOTP Server:	192.168.1.1
TFTP Server:	Not Specified
DHCP Expiration Time:	00:23:15 (Days:Hours:Minutes)
WINS Server:	Not Specified
Preferred DNS Address:	212.50.160.100
Alternate DNS Address:	213.249.130.100

TCP/IP(v6)

Status:	Ready
Link-Local Address:	FE80::1A60:24FF:FEC7:F9E3
Stateless (from Router):	Not Specified
Stateful (from DHCPv6):	Not Specified
Preferred DNS Address:	Not Specified
Alternate DNS Address:	Not Specified

Network Identification

Host Name:	NPIC7F9E3
Domain Name (IPv4/IPv6):	
Domain Name (IPv6 only):	
Bonjour Service Name:	HP Color LaserJet M452dn (C7F9E3)
Bonjour Domain Name:	NPIC7F9E3.local.
Bonjour Highest Priority Service:	IPP Printing

Network Hardware Configuration

Installing a family printer in 2017

How would you design this?

Is this a good UI?

How do we know?

Could we improve it?

General

Security and login

Privacy

Timeline and taggi...

Blocking

Language

Notifications

Mobile

Public posts

Apps

Ads

Payments

Support Inbox

Videos

Privacy Settings and Tools

Your activity

Who can see your future posts?

Friends

Edit

Review all your posts and things you're tagged in

Use Activity Log

Limit the audience for posts you've shared with friends of friends or Public?

Limit Past Posts

How people can find and contact you

Who can send you friend requests?

Everyone

Edit

Who can see your friends list?

Public

Edit

Who can look you up using the email address you provided?

Everyone

Edit

Who can look you up using the phone number you provided?

Everyone

Edit

Do you want search engines outside of Facebook to link to your Profile?

Yes

Edit

About

Create ad

Create Page

Developers

Careers

Privacy

Cookies

AdChoices

Terms

Help

Facebook © 2017

English (UK)

English (US)

Română

Magyar

Italiano

Español (España)

Français (France)

Deutsch

Türkçe

Português (Brasil)

العربية

Facebook privacy in 2017

How would you design this?

Is this a good UI?

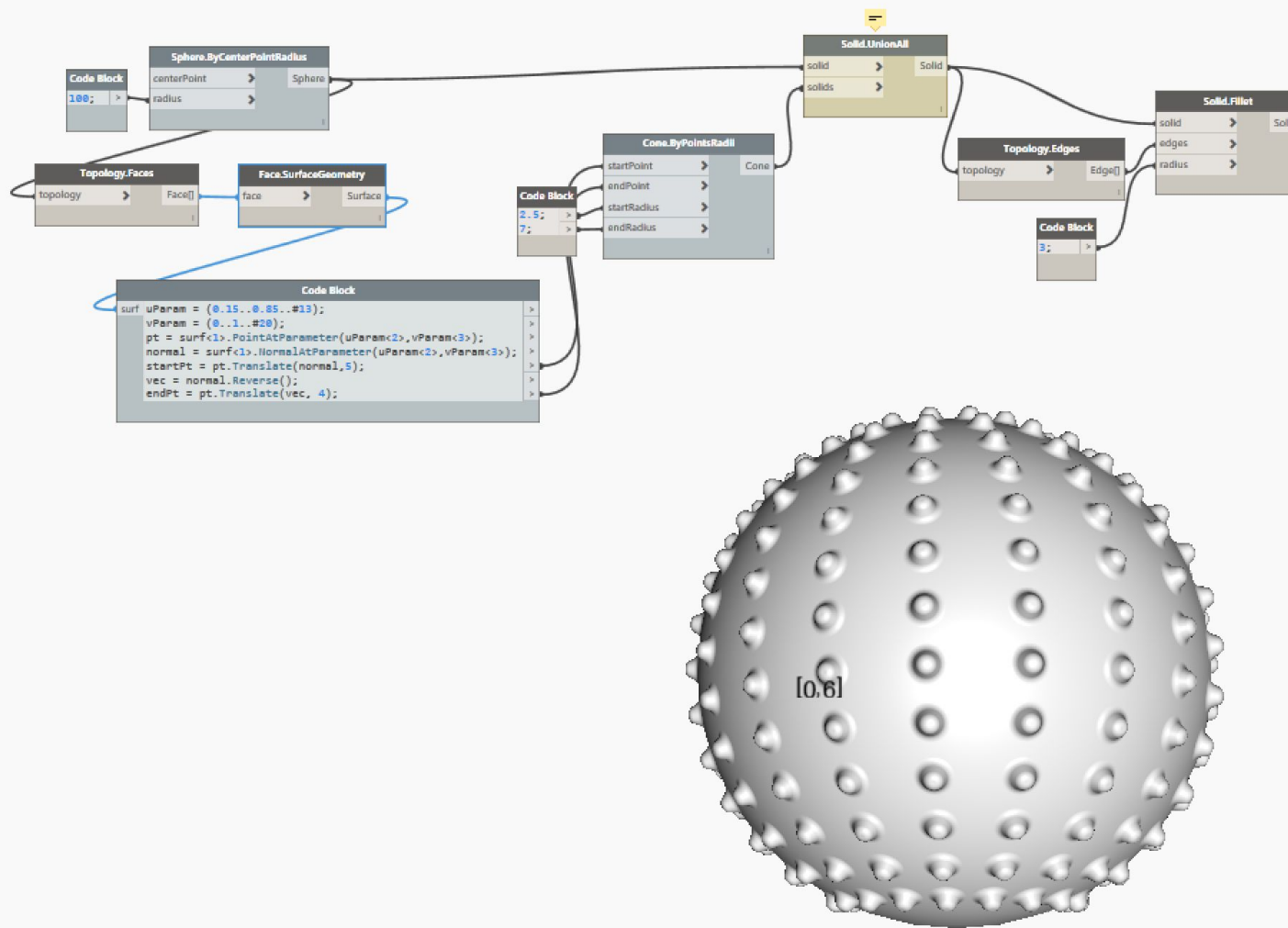
How do we know?

Could we improve it?

Visual Programming in 2017

How would you design this?

Is this a good programming language?



Claude

New chat

Search

Chats

Projects

Artifacts

Code

Recents

Certificate verification error in curl

CPU timestamp counter explained

Choosing a Python package mana...

Choosing a Python package mana...

Whimsical names for commentato...

GCP instance startup script locati...

GCP instance startup script location

Share

I have a GCP instance that seems to be running a specific shell script on startup, where is that likely to be? I/we created it in November.

For a GCP instance running a startup script, there are a few places to check:

1. **Instance metadata** - Most likely location. Check the instance's metadata for a `startup-script` or `startup-script-url` key:

```
bash
gcloud compute instances describe INSTANCE_NAME \
  --zone=ZONE \
  --format="value(metadata.items[startup-script])"
```

2. **Instance template** (if you're using one):

```
bash
gcloud compute instance-templates describe TEMPLATE_NAME \
  --format="value(properties.metadata.items[startup-script])"
```

3. **Local startup scripts directory** on the VM itself:

- `/etc/rc.local` (legacy)
- Systemd service files in `/etc/systemd/system/`
- Scripts in `/etc/init.d/`

4. **Cloud Storage bucket** - If you used `startup-script-url`, the actual script would be stored in a GCS bucket

5. **Terraform/IaC configuration** - Since you mentioned creating it in November, check your Terraform configs for a `metadata_startup_script` parameter in the `google_compute_instance` resource

The metadata approach is the standard GCP way. You can view it in the Cloud Console too: go to Compute Engine → VM instances → click your instance → scroll to "Custom metadata" section.

What type of instance is it, and do you remember how you created it (Console, gcloud, Terraform)?

Reply...

Sonnet 4.5

Luke

Pro plan

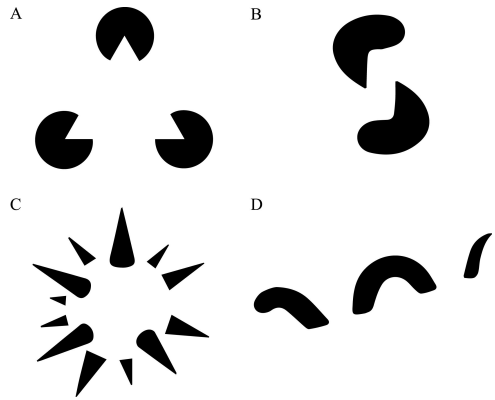
Programming in 2026

How would you design this?

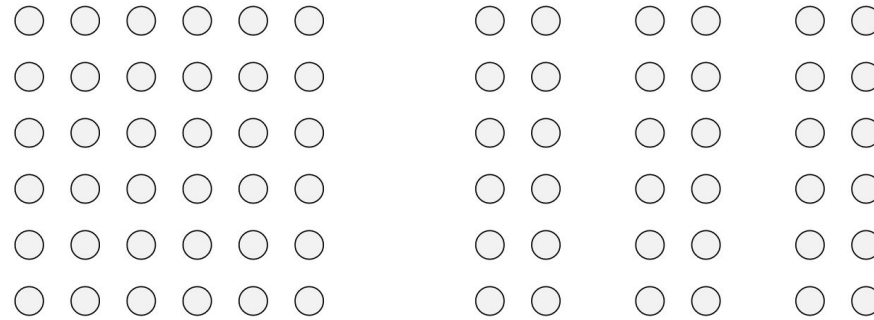
Is this a good programming language?

Theories give a *critical perspective*

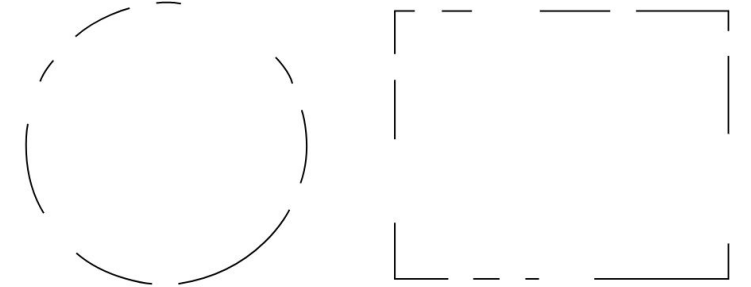
Reminder of a theory: Gestalt theory of perceptual organisation



Continuity

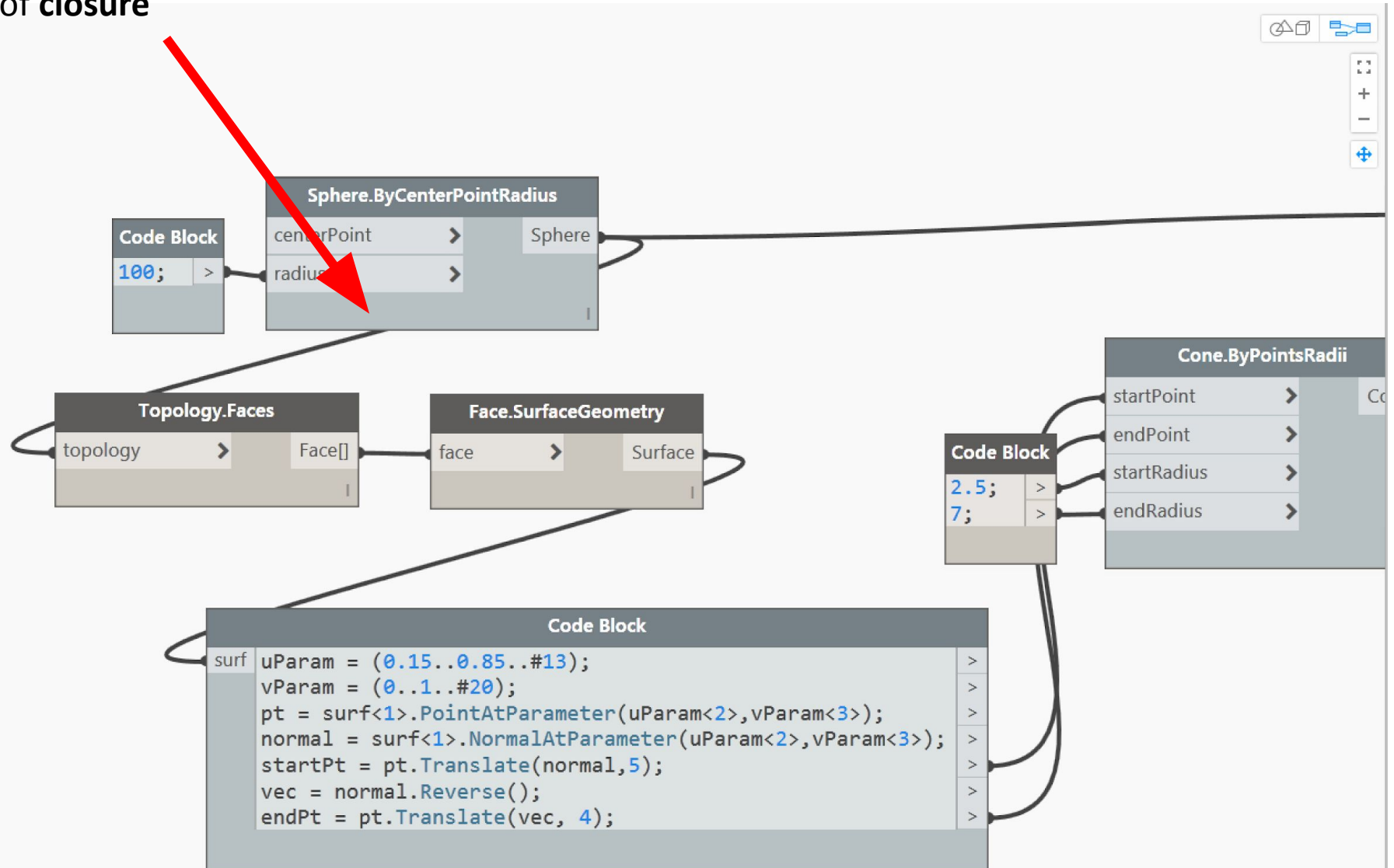


Similarity

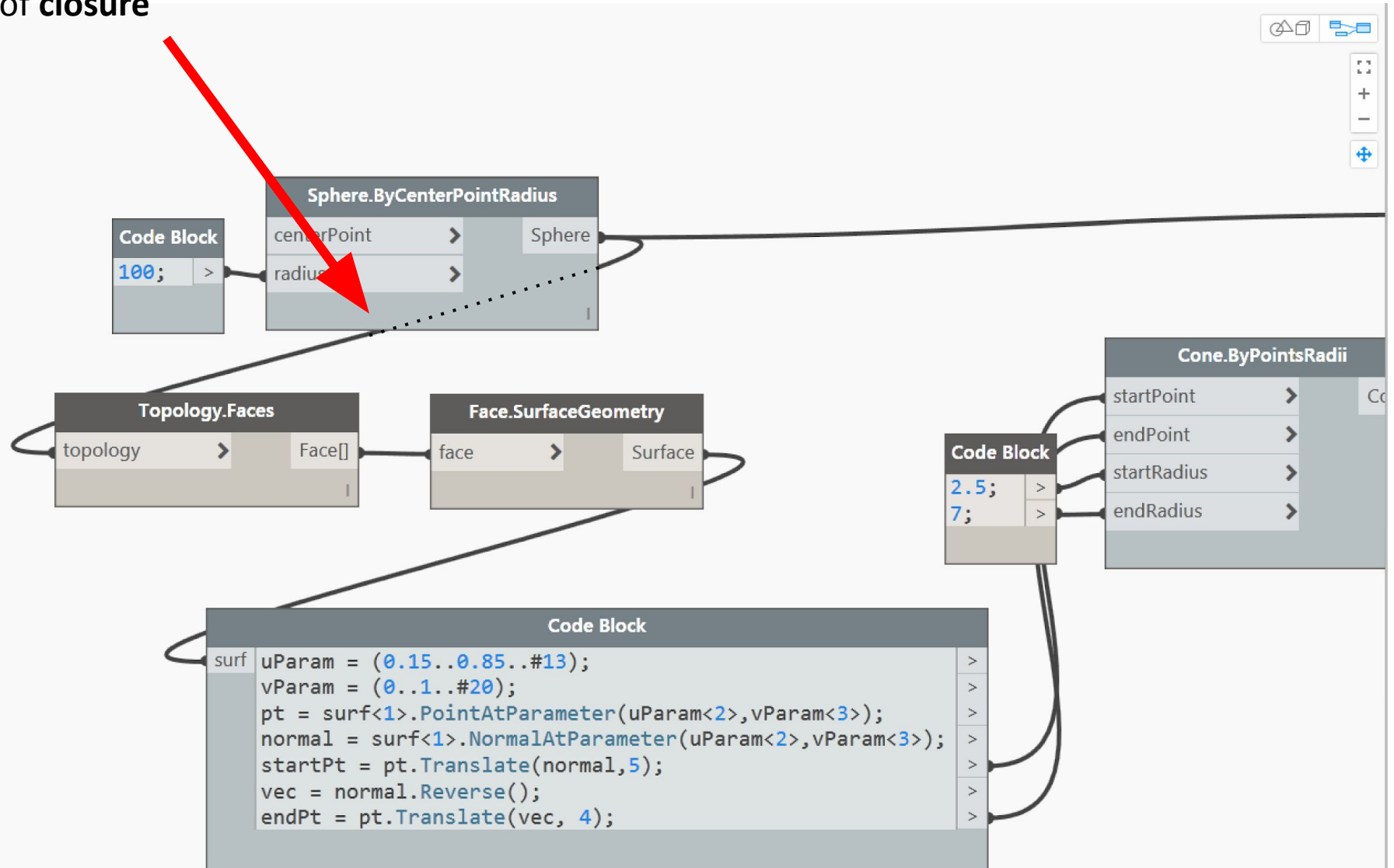


Closure

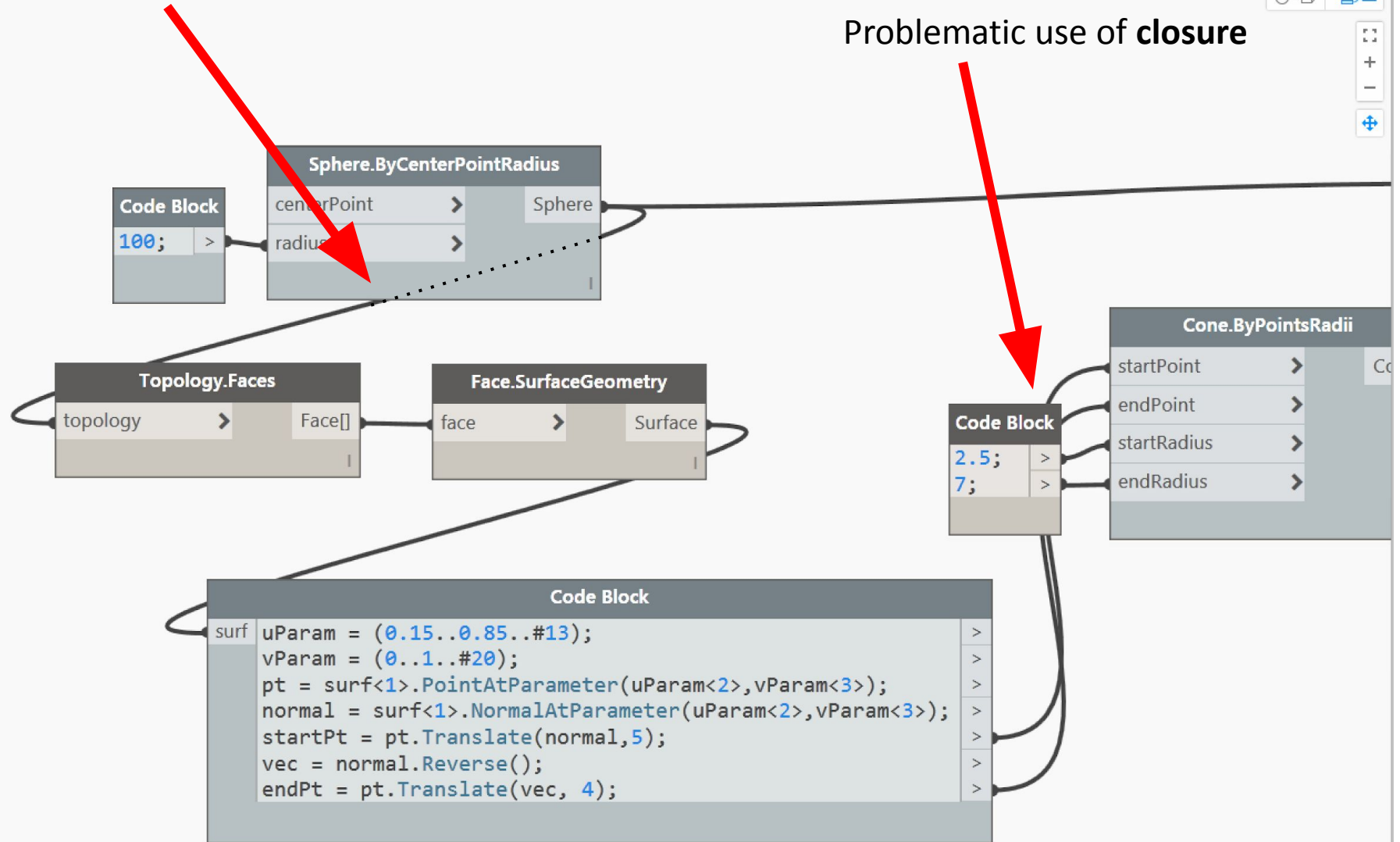
Use of closure



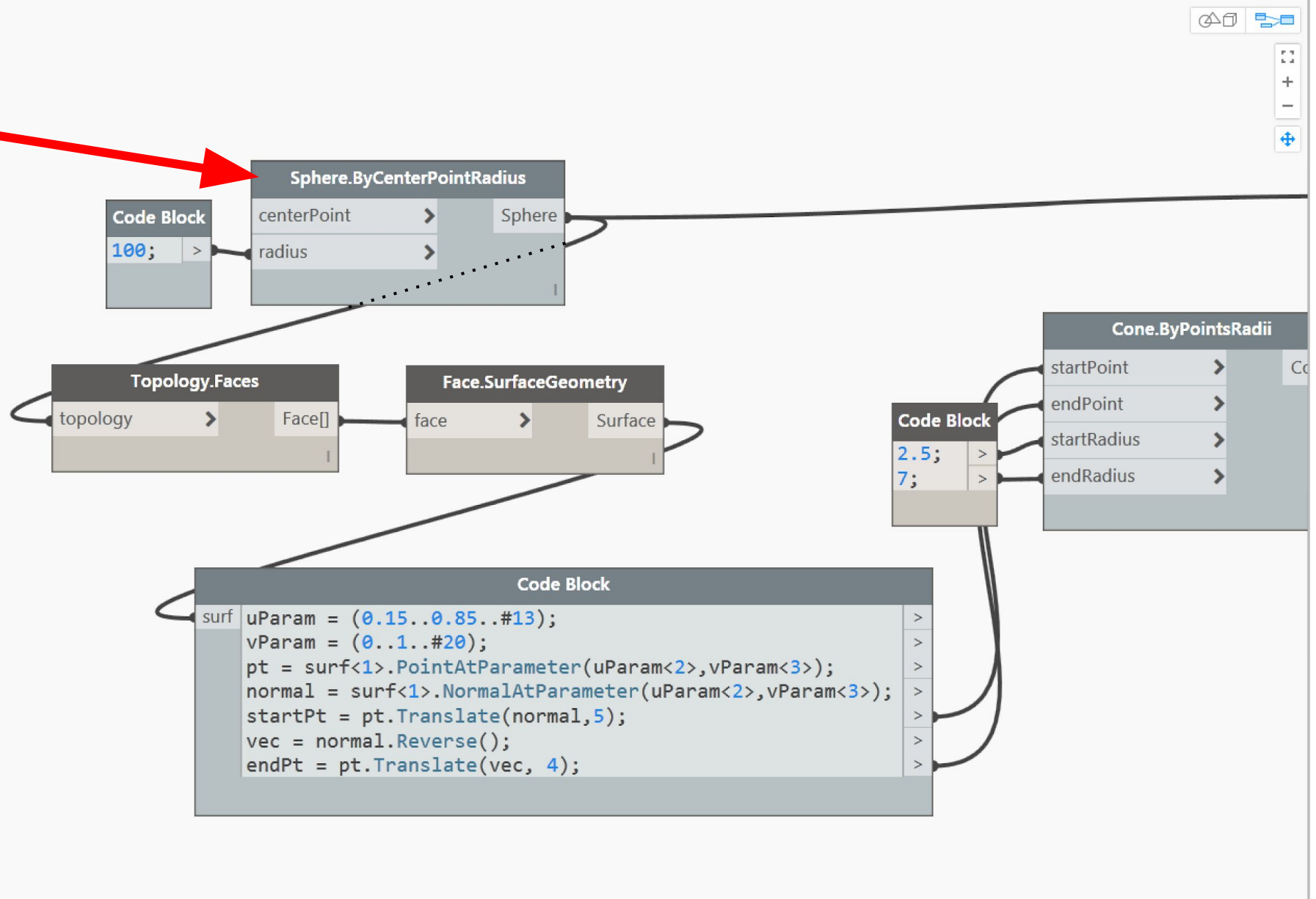
Use of closure



Use of closure

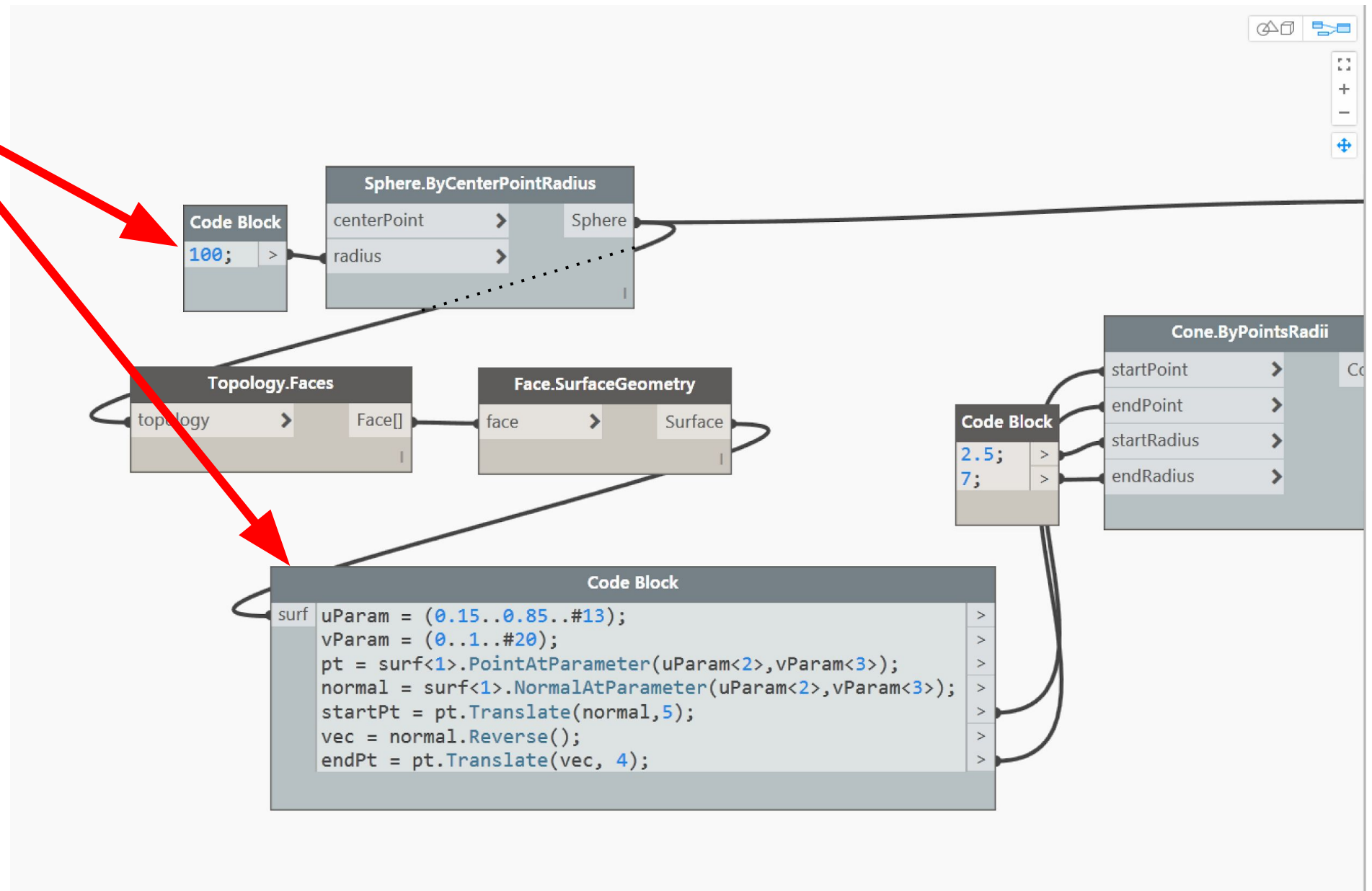


No use of **continuity**



Problematic Similarity

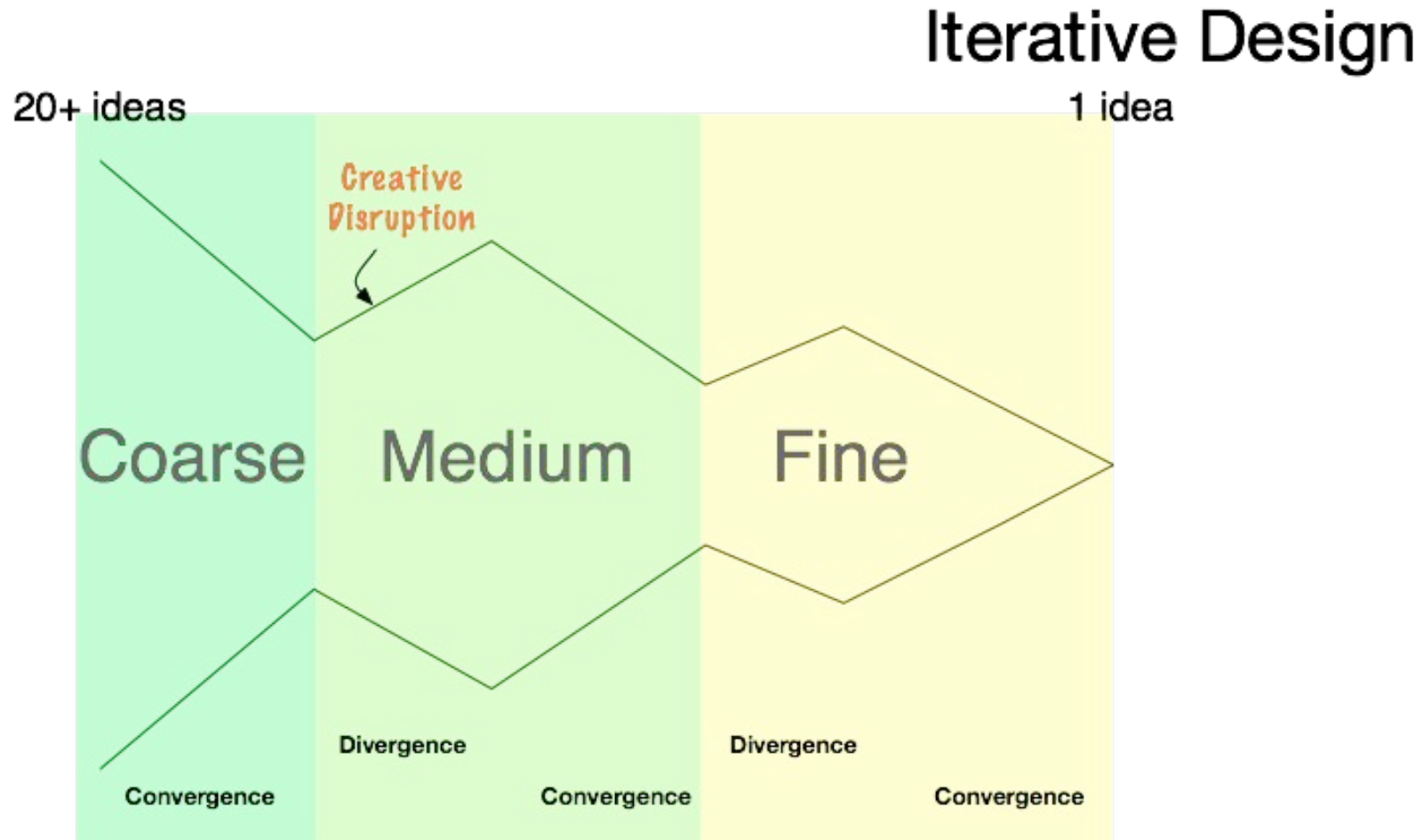
Do users think these
are the same?



Summary of gestalt theory application

- Took a candidate design (Dynamo UI)
 - Predicted some properties that probably work well
 - Predicted some properties that might cause problems
- Over the course of the lectures you'll see many theories like this
- How do we make use of critique?

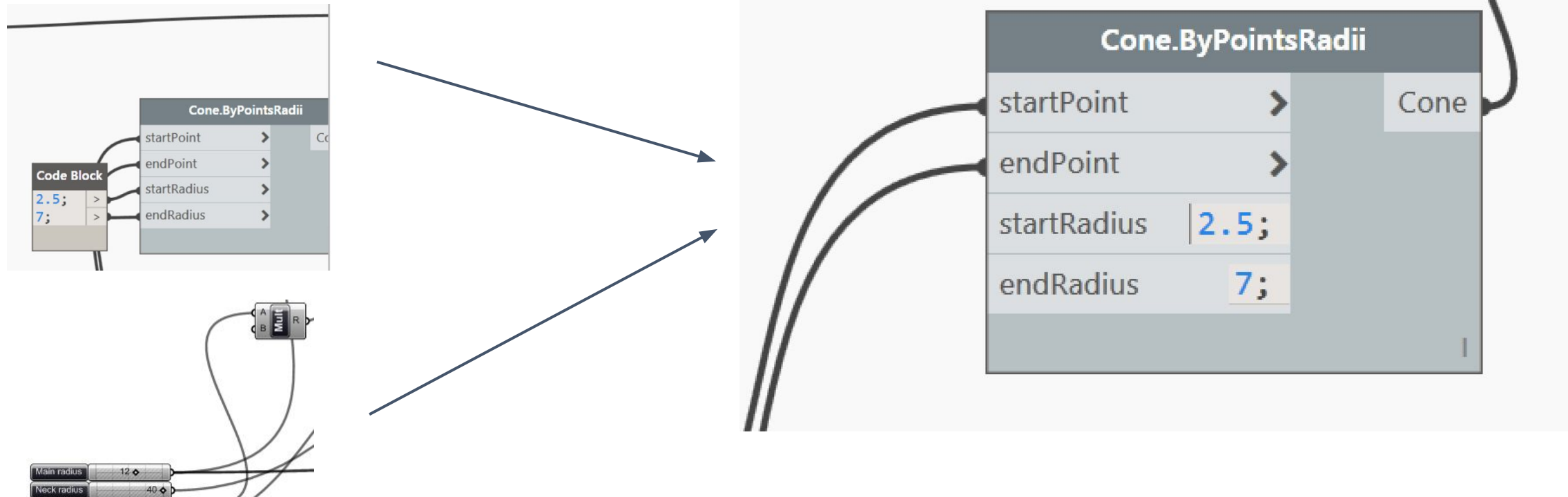
Critique your way to a design



Derived from Pugh '56

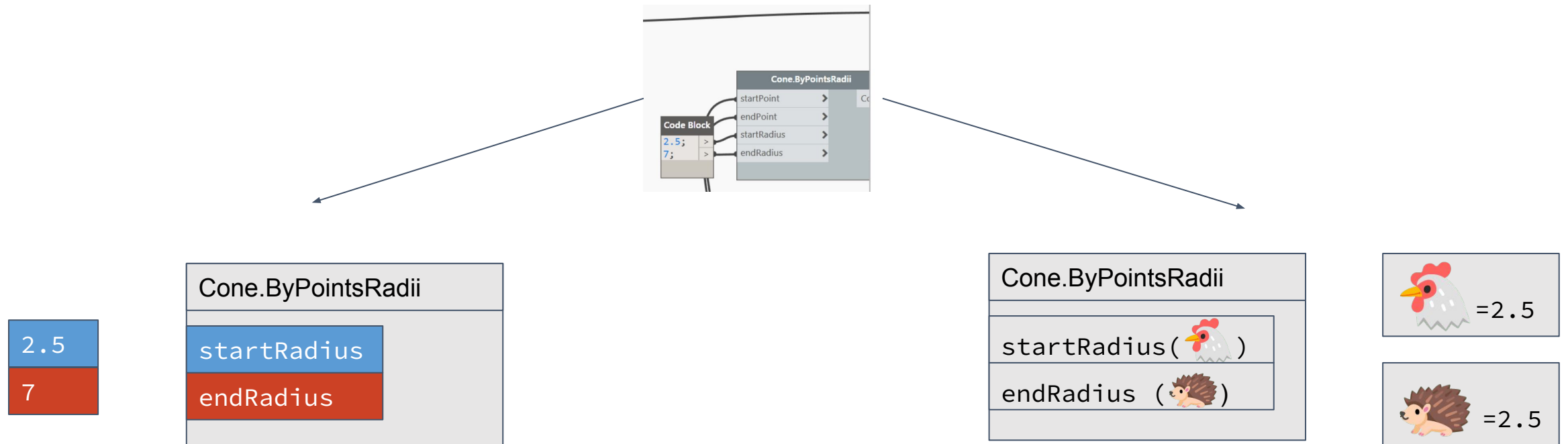
Example of convergence

- Merge features from two candidate designs to produce a better one
- Discard ideas that poorly fit the desired outcome



Example of divergence

- Generate new designs from existing one
- Use any creative technique, e.g. 'gestalt swapping', 'reduction to absurdity' or exploring metaphors
(e.g. what happens if we replace connectedness with similarity)



Why theory in HCI?

- We've interested in making interaction with computers faster, more productive, more creative, more social, more fun, somehow 'better'.
- Theories give us ways of criticising proposed designs and toolkits for inventing new ones

Three waves of HCI

- First wave (1980s):
 - Theory from Human Factors, Ergonomics and Cognitive Science
- Second wave (1990s):
 - Theory from Anthropology, Sociology and Work Psychology
- Third wave (2000s):
 - Theory from Art, Philosophy and Design



Apollo-Soyuz controls 1975

How would you design this?

Is this a good UI?

How do we know?

Could we improve it?

First wave: HCI as engineering “human factors”

- The “user interface” (or MMI “man-machine interface”) is a specialist module, designed independently of the main system.
- Design goal is efficiency (speed and accuracy) for a human operator to achieve well-defined functions.
- Use methods from cognitive science and ergonomics to model users’ perception, decision and action processes and predict usability.



An information system

How would you design this?

Is this a good UI?

How do we know?

Could we improve it?

Second wave: HCI as social system

- The design of complex systems is a socio-technical experiment
 - Take account of other information factors including conversations, paper, and physical settings
- Study the context where people work
 - Use Ethnography and Contextual Inquiry to understand other ways of seeing the world
- Other stakeholders are integrated into the design process
 - Prototyping and participatory workshops aim to empower users and acknowledge other value systems



Blood bag radio

How would you design this?

Is this a good UI?

How do we know?

Could we improve it?

Third wave: HCI as culture and experience

- Ubiquitous computing affects every part of our lives
 - It mixes public (offices, lectures) and private (bedrooms, bathrooms)
- Outside the workplace, efficiency is not a priority
 - Usage is discretionary
 - User Experience (UX), includes aesthetics, affect,
- Design experiments are speculative and interpretive
 - Critical assessment of how this is meaningful

Specialist topics not covered here:

- Graphics and VR - elsewhere in CS Tripos
- Digital media studies - Cambridge Digital Humanities
- Game design - Anglia Ruskin University
- Social network analysis - elsewhere in CS Tripos
- Computer music - elsewhere in CS Tripos, Centre for Music & Science
- Security - elsewhere in CS Tripos
- Educational technology - Faculty of Education
- Information Systems - Judge Business School

Alternative perspectives

- Positive computing (e.g. Calvo & Peters 2014)
 - Wellbeing, flow, empathy, mindfulness, altruism
- Inclusion and accessibility (e.g. CWUAAT #1-9)
 - physical and sensory capabilities, ageing, low income and human rights
- Feminist utopianism (e.g. Bardzell 2010)
 - Diagnostic critique of hegemonic research and practice, combined with practice-led participatory processes of anticipation that amplify marginalized voices

Supervisions

- 2 supervisions after lecture 4 and lecture 8, recommend completing all the lectures before the last supervision

Textbooks

- Preece, Sharp & Rogers *Interaction Design: Beyond human-computer interaction* - (5th edition 2019)
 - Practical professional advice, with good summaries of relevant theories and research methods
 - Any edition is useful, and there are many copies in Cambridge libraries
- Carroll (Ed.) *HCI Models, Theories and Frameworks: Toward a multidisciplinary science* 2003
 - Expert introductions to the different theoretical traditions of the first and second waves (noting that the third wave is more practice-based, beyond purely academic theory)

Toward original research

We look at user interfaces and software systems through the lens of programming languages. We think this approach illuminates a lot of the important properties of the system.

This is an advanced perspective, but especially relevant to Cambridge students, and to future innovation (the GUI was originally a programming language!)

See Alan's new book *Moral Codes: Designing alternatives to AI* for an extended version of this argument.

For now, we'd love you to give us examples of systems you'd like to talk with us about during the course.

Which systems shall we talk about?

- ChatGPT
- Outlook
- Teams
- Panopto
- Moodle
- Windows desktop
- VSCode
- Tesla FSD

Choices from 2025

- Electronic Patient Records
- Note taking tablet
- Music composition
- XTwitter
- Tax payment system
- 3D printer
- Isabel programming language