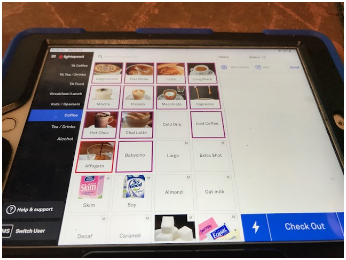
On the “surjectivity” of lenses

Michael Johnson (with Robert Rosebrugh)

BX2021, Bergen

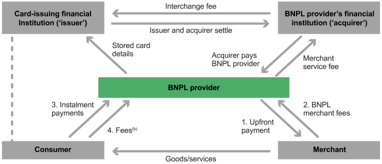
Bx are ubiquitous



Bx are ubiquitous

Bx are ubiquitous

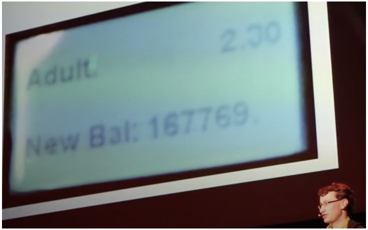
BNPL

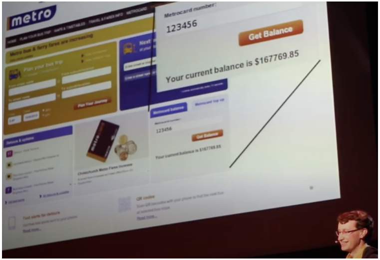


Fintech

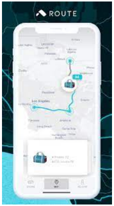
Brokerage

Bx are ubiquitous





Bx are ubiquitous — non-example



.

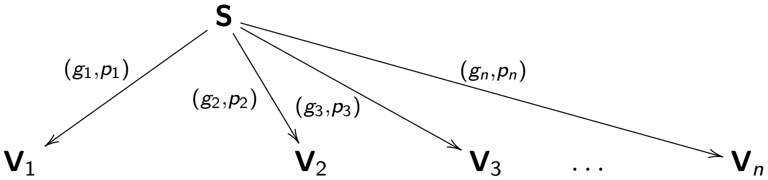
Bx are ubiquitous

- Restaurants
- Trains
- BNPL, Fintech, Brokerage more generally . . .
- Public transport smart cards
- Non-example: Parcel tracking

Bx are ubiquitous and so . . .

- Many people are writing (relatively routine) bidirectional, or indeed multidirectional, transformations (although the vast majority of them have probably never heard of "Bx")
- Meanwhile, we Bx people are researchers, and we work on challenging problems like synchronising different software engineering artifacts (cf Perdita's Inaugural Lecture)
- This is a talk about the software engineering of relatively routine multi-directional transformations, and a particular bidirectional transformation between two software engineering models (used respectively for design and implementation)

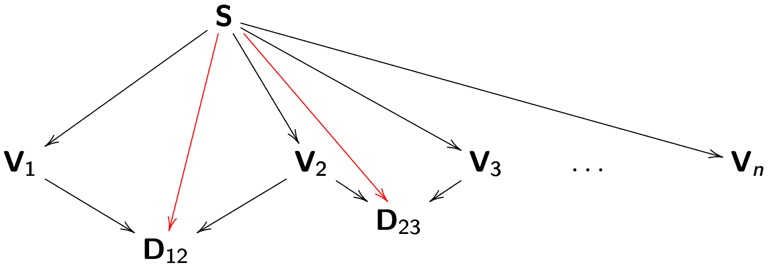
Design


S
(g_1, p_1)
(g_2, p_2)
(g_3, p_3)
(g_n, p_n)
$\mathbf{V}_1$
$\mathbf{V}_2$
$\mathbf{V}_3$
$\dots$
$\mathbf{V}_n$

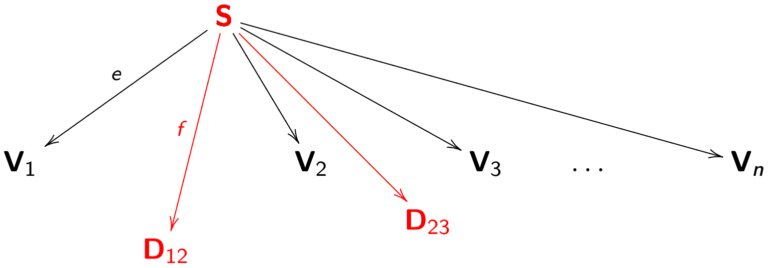
Implementation

$$
\begin{tikzcd}
\mathbf{V}_1 \arrow[dr] & & \mathbf{V}_2 \arrow[dl] \arrow[dr] & & \mathbf{V}_3 \arrow[dl] & \cdots & \mathbf{V}_n \\
& \mathbf{D}_{12} & & \mathbf{D}_{23} & & &
\end{tikzcd}
$$

Implementation to Design (calculate "limit")



Design to Implementation (complete the triangles)



Design to Implementation (complete the triangles)

Given e and f, determine w

$$
\begin{tikzcd}
 & \mathbf{S} \arrow[dl, "e"'] \arrow[dr, "f"] & \\
\mathbf{V}_1 \arrow[rr, "w"'] & & \mathbf{D}_{12}
\end{tikzcd}
$$

Some (false) folk theorems

1. Asymmetric d-lenses factorise uniquely: $\overset{e}{\swarrow} \cdot \overset{f}{\searrow}$

   Given a span of lenses (e, f) that is known to be two sides of a commutative triangle of lenses, the third side (a lens w) is derivable (uniquely) from the other two
2. Well-behaved set-based asymmetric lenses are surjective: The Get function is necessarily surjective
3. Assymetric d-lenses are (or at least should be) surjective: The Get functor is necessarily surjective on both objects and arrows (at least if one ignores irrelevancies)

The main theorem

There is a proper orthogonal factorisation system on the category of asymmetric d-lenses

Summary

- Relatively routine multidirectional transformations are being built ‘even as we speak’
- There is usually a bidirectional transformation between their implementation and design models
- That bidirectional transformation is easy one way (take the “limit”), but surprising the other way (complete the triangles)
- The surprise is resolved by realising that
  1. Engineers frequently treat their lenses as surjective
  2. There is a proper orthogonal factorisation system on the category of lenses that justifies that

Factorisation Systems (Appendix)

A *proper orthogonal factorisation system* on a category **C** consists of two classes of arrows of **C** called the *left class*, denoted $\mathcal{L}$, and the *right class*, denoted $\mathcal{R}$, such that

1. arrows in $\mathcal{L}$ are epis in **C** and arrows in $\mathcal{R}$ are monos in **C**,
2. every arrow a of **C** factors as $a = me$ when $e \in \mathcal{L}$ and $m \in \mathcal{R}$ and
3. for any commutative square as shown with vertical arrows e and m from $\mathcal{L}$ and $\mathcal{R}$ respectively, there exists an arrow w of **C** (necessarily unique) making the two triangles commute.

$$\begin{tikzcd}
\mathbf{S} \arrow[r, "f"] \arrow[d, twoheadrightarrow, "e"'] & \mathbf{D}_{12} \arrow[d, tail, "m"] \\
\mathbf{V}_1 \arrow[r] \arrow[ur, dashed, "w"] & \mathbf{K}
\end{tikzcd}$$