Meeting a Fanclub: A Lattice of Generic Shape Selectors[1]

Roland Backhouse
School of Computer Science, University of Nottingham, UK
rcb@cs.nott.ac.uk

Richard Bird
Computer Laboratory, University of Oxford, UK
Richard.Bird@comlab.ox.ac.uk

Paul Hoogendijk
Philips Corporate Technologies, Eindhoven, The Netherlands
Paul.Hoogendijk@philips.com

Abstract

The "fan" of a datatype F is a relation that holds between a value x and an arbitrary F structure in which the only stored value is x. Fans make precise the notion of the shape of a data structure. We formulate two different representations of shape selectors and exploit the properties of fans to prove that the two representations are order isomorphic and that shape selectors are closed under set intersection. For arbitrary datatypes F, G and H, we consider six different ways of composing their fans in order to construct F structures of G structures of H structures; each of the six imposes a different requirement on the shape of the substructures. We catalogue the relation between different combinations of the constructions. We apply the result to a problem that arose in a generic theory of dynamic programming concerning the shape properties of a natural transformation from G structures to FG structures.

Categories and Subject Descriptors D,F [*D.1,F.3*]: D.1.1,F.3.3

General Terms Algorithms, Languages, Theory

Keywords datatype, collection type, relation algebra, allegory, funtional programming, relational programming, dynamic programming

1. Introduction

Consider the following problem about the shape properties of three arbitrary datatypes F, G and H. Suppose θ is a natural transformation taking G structures to FG structures with the property that the result of θ is an F structure of G structures all of which have the same shape. Consider the instance θH (which transforms a GH structure to an FGH structure). Show that if θH is applied to a G structure of H structures all of the same shape, the result is an F structure of GH structures all of which have the same shape.

A problem like this arose in a recent study of dynamic programming [6]. It is an example of a problem in (datatype-)generic programming; the challenge is to develop formalisms in which such problems can be readily stated and solved, and in a way that is appropriate to the needs of the practising programmer. Such a formalism is furnished by the relational theory of datatypes [1, 3, 7, 9, 11]). The above problem acted as a catalyst for us to investigate, in much greater depth than before, the properties of shape-preserving transformations; the results of the investigation are reported in this paper.

In our relational theory, datatypes (also known as "collection" types — types like lists and binary trees) are modelled by "relators" with "membership".

The notion of a relator plays the role in allegory theory —which for our purposes is essentially the theory of binary relations on sets— of the notion of a functor in category theory [7]. Just as functors model the structural properties of datatypes in conventional functional programming, relators model the structural properties of datatypes in generic programming.

The formal notion of membership models the idea that datatypes are mechanisms for storing data. The membership relation of a datatype F, which we denote by $\mathsf{mem}.F$, is a binary relation between values and F structures which holds when the value is stored in the F structure. For example, the relation $\mathsf{mem.List}$ holds between a value x and a list xs if x is a member of xs (according to the standard notion of list membership). Note that relations (as opposed to functions) are essential to the theory for the simple reason that membership is a relation.

Shape properties of datatypes are formulated using the notion of a "fan" [10, 9] (called a "generator" in [2] where it was first introduced). The fan of a datatype F is a binary relation between F structures and values; it holds when every member of the F structure is equal to the given value. Another way of viewing the fan, $\mathsf{fan}.F$, of a datatype F is as a non-deterministic program that, given a so-called *seed*, constructs an arbitrary F structure in which the only stored value is the seed.

Membership and fans are both natural transformations. Specifically, $\mathsf{mem}.F$ transforms an F structure to an Id structure (where Id denotes the identity relator); it is "natural" ("polymorphic" in the jargon of functional programming) in that it is not dependent on the type of values stored in the structure. We write this as $\mathsf{mem}.F : \mathsf{Id} \hookleftarrow F$. Conversely, $\mathsf{fan}.F$ transforms an Id structure to an F structure; that is, $\mathsf{fan}.F : F \hookleftarrow \mathsf{Id}$. The instance $(\mathsf{fan}.F)G$ has as seed a G structure, which is transformed to an FG structure by constructing an arbitrary F structure, and copying the seed into all the storage locations. In particular, $(\mathsf{fan}.F)G$ constructs an FG structure in which all the stored G structures have the same shape.

Given this basis, it is relatively straightforward to formulate the problem we used to introduce this paper. It is about three datatypes F, G and H, and the relationship between a natural transformation of type $FG \hookleftarrow G$ and the fans of the datatypes.

There are six different ways that the fans of the datatypes F, G and H can be composed to form a natural transformation of type $FGH \hookleftarrow \mathsf{Id}$. This is the "fanclub" in the title of the paper. We explore the relationship between combinations ("meets") of all six. That is, we formulate and prove properties like: an FGH structure with the property that all the G structures have the same shape and all the H structures have the same shape also has the

WGP'09, August 30, 2009, Edinburgh, Scotland, UK.

property that all GH structures have the same shape. Section 3 introduces the fanclub and gives examples, whilst the main results are recorded in section 5. Section 4 studies shape properties of natural transformations in general (as opposed to the particular properties of fans). We return to our introductory problem in section 4.2.

2. Preliminaries

This section summarises the formal basis for our work. For more details see [7, 10, 9, 5]. In brief, we model relations as the arrows of an allegory, and a datatype (like List) as a relator (section 2.1) with membership (section 2.5).

2.1 Allegories and Relators

An *allegory* [8] is a category with additional structure, the additional structure capturing the most essential characteristics of relations.

Being a category means that for every object A there is an identity arrow id_A, and every pair of arrows $R : A \leftarrow B$ and $S : B \leftarrow C$, with matching source and target, can be composed: $R \circ S : A \leftarrow C$. Composition is associative and has id as a unit.

The direction of the arrows is, of course, irrelevant to the development of the theory. We use left-pointing arrows to suggest the interpretation of arrow R of type $A \leftarrow B$ as a nondeterministic program with outputs of type A and inputs of type B; the expression $(x,y) \in R$ is read as: x is an "output" of the relation R given "input" y. In other words, arrows in an allegory are interpreted as input-output relations, where the input is on the right of the arrow and the output is on the left. This is in line with function application where the order of writing is function applied to argument.

The additional axioms are as follows. First of all, arrows of the same type are ordered by the *partial order* $\subseteq$ and composition is monotonic with respect to this order. Secondly, for every pair of arrows $R, S : A \leftarrow B$, their *intersection* (*meet*) $R \cap S$ exists and is defined by the customary universal property. Finally, for each arrow $R : A \leftarrow B$ its *converse* $R^{\cup} : B \leftarrow A$ exists. The converse operator is defined by the requirements that it is its own Galois adjoint, and is contravariant with respect to composition. All three operators of an allegory are connected by the *modular law*, also known as Dedekind's law [14]: for all R, S and T (of appropriate type),

$$R \circ S \cap T \quad \subseteq \quad R \circ (S \cap R^{\cup} \circ T) \ .$$

(Note that composition has precedence over intersection. However, we often parenthesise expressions more than strictly necessary in order to assist the reader unfamiliar with the convention. We try to space formulae in order to suggest to the eye the appropriate parsing.) We also apply the law in its converse form:

$$S \circ R \cap T \quad \subseteq \quad (S \cap T \circ R^{\cup}) \circ R \ .$$

Generally, every law in allegory theory has a symmetric "converse form"; we often state just one form but use both.

The standard example of an allegory is Rel, the allegory with sets as objects and relations as arrows. With this allegory in mind, we refer to the arrows of an allegory as "relations".

A *relator* is a monotonic functor that commutes with converse. We generally use F, G and H to denote relators. Application of F to R is denoted by FR. The identity relator is denoted by Id. For given object A, K_A denotes the constant relator — the relator which maps objects to A and arrows to id_A.

2.2 Universal Relations, Units and Division

The allegory Rel has more structure than we have captured so far with our axioms. In full, Rel is a unitary, tabulated, locally complete, division allegory. For full discussion of these concepts see [8] or [7]. In this paper, very little of this structure is required explicitly (although some of the structure does underlie some of the properties we exploit). We detail just what we need.

We assume that, for each pair of objects A and B, there is a smallest and a largest relation of type $A \leftarrow B$, which we denote by $\perp\!\!\!\perp_{A,B}$ and $\top\!\!\!\top_{A,B}$, respectively. We omit the subscripts when the information they provide is not essential. The symbol "$\perp\!\!\!\perp$" is pronounced "bottom" and the symbol "$\top\!\!\!\top$" is pronounced "top". The interpretation of $\perp\!\!\!\perp_{A,B}$ is the empty relation between values of type A and values of type B; it is the zero of composition. The interpretation of $\top\!\!\!\top_{A,B}$ is the universal relation between values of type A and values of type B.

We also assume the existence of a *unit* object, denoted by "$\mathsf{1}$". Formally, $\mathsf{1}$ is such that id_1 is the largest relation of its type — that is, $\mathsf{id}_1 = \top\!\!\!\top_{1,1}$ — and, for every object A there is a total relation $!_A$ of type $\mathsf{1} \leftarrow A$. An immediate consequence of this definition is that $!_A$ is a *function*. That is, it is "simple" (i.e. $!_A \circ !_A{}^{\cup} \subseteq \mathsf{id}_1$) and —by definition— "total" (i.e. $!_A{}^{\cup} \circ !_A \supseteq \mathsf{id}_A$). In general, relation R of type $A \leftarrow B$ is said to be *simple* if $R \circ R^{\cup} \subseteq \mathsf{id}_A$ and *total* if $R^{\cup} \circ R \supseteq \mathsf{id}_B$. (This terminology also reflects the interpretation of relation R of type $A \leftarrow B$ as a non-deterministic program mapping inputs of type B to outputs of type A.) A second consequence is that $!_A = \top\!\!\!\top_{1,A}$. (The simple proof is omitted.)

Finally, we assume the existence of the so-called *division* operator "$\backslash$". Specifically, we have the following Galois-connection. For all $R : A \leftarrow B$, $S : B \leftarrow C$ and $T : A \leftarrow C$,

$$R \circ S \subseteq T \quad \equiv \quad S \subseteq R \backslash T \ .$$

Note that $R \backslash T : B \leftarrow C$.

2.3 Domains

In addition to the source and target of a relation it is useful to know their left and right "domains". The *left domain* of a relation R is that subset $R{<}$ of id_A, where A is the target of R, defined by the Galois connection:

$$R \subseteq X \circ \top\!\!\!\top_{A,B} \quad \equiv \quad R{<} \subseteq X \quad \text{for each } X \text{ such that } X \subseteq \mathsf{id}_A \ . \qquad (1)$$

The *right domain* of $R : A \leftarrow B$, which we denote by $R{>}$, is the left domain of $R^{\cup}$. The *complement of the right domain* of R is denoted by $R{>}\!\bullet$; it is the largest X such that $X \subseteq \mathsf{id}_B$ and $R \circ X \subseteq \perp\!\!\!\perp_{A,B}$.

The interpretation of the left domain of a relation R of type $A \leftarrow B$ is the set of all a in A such that $(a,b) \in R$ for some b in B.

A relation R of type $A \leftarrow A$, for some A, that is a subset of id_A is called a *coreflexive* (or *monotype*). An important property is that composition of coreflexives is the same as intersection (that is, for coreflexives R and S, $R \circ S = R \cap S$) and so is commutative. Also, composition with a coreflexive (on the left or right) distributes through intersection. Domains are coreflexives.

We frequently use the property that

$$(R \circ S{<}){<} \quad = \quad (R \circ S){<} \qquad (2)$$

(for all R and S); the accompanying hint is "domains". We also use the fact that the domain operators are monotonic, but we rarely state this explicitly in the hints. A closed formula for $R{<}$ is $\mathsf{id}_A \cap (R \circ R^{\cup})$.

Relations of type $\mathsf{1} \leftarrow A$ are called *right conditions*. There is an important order-preserving isomorphism between coreflexives and right conditions given by the function mapping coreflexive X of type $A \leftarrow A$ to right condition $!_A \circ X$, and the right domain operator mapping right condition C to $C{>}$. Specifically, $(!_A \circ X){>} = X$ and $!_A \circ C{>} = C$. Symmetrically, there is an order-preserving isomorphism between coreflexives and left conditions (where a left condition is a relation of type $A \leftarrow 1$).

Typically, the domain operators do not distribute through intersection. However, when applied to conditions, they do. That is, if C and D are both left conditions of the same type,

$$(C \cap D)< \;=\; C< \cap D< \ .$$

Because of the isomorphism between coreflexives and (left or right) conditions, it is always possible to interchange between the two. Expressed in terms of left conditions, property (2) then becomes trivial — it is the property that composition is associative. For this and similar reasons, conditions are sometimes preferable to coreflexives. However, in the relational theory of datatypes there is a very cogent argument for using coreflexives; it is the fact that relators commute with the domain operators. That is, for all relators F and all R,

$$(FR)< \;=\; F(R<) \ .$$

We usually use this rule silently: we write $FR<$ and exploit the ambiguity in the operator precedence.

2.4 Natural Transformations

We define a collection of relations θ indexed by objects to be *a natural transformation of type* $F \hookleftarrow G$ for relators F and G if, for all types A, $\theta_A : FA \leftarrow GA$, and, for all $R : A \leftarrow B$,

$$FR \circ \theta_B \;\supseteq\; \theta_A \circ GR \ . \tag{3}$$

(See [9, 5] for a detailed discussion of why containment rather than equality of relations is the appropriate way to define a natural transformation in an allegory.) The composition of natural transformation θ of type $F \hookleftarrow G$ and φ of type $G \hookleftarrow H$ is denoted by $\theta \circ \varphi$. The subscripts on natural transformations will be omitted when the information they provide is not essential. In the process, operations like application of relators will be silently "lifted" to natural transformations in the same way as we have overloaded the composition operator above.

If θ is a natural transformation of type $F \hookleftarrow G$ and H is a relator, θH is a natural transformation of type $FH \hookleftarrow GH$ and $H\theta$ is a natural transformation of type $HF \hookleftarrow HG$. Specifically, $(\theta H)_A$ is (by definition) θ_{HA} and $(H\theta)_A$ is (again by definition) $H(\theta_A)$.

A natural transformation is said to be *proper* if the inclusion in (3) can be strengthened to an equality. That is, θ is a *proper natural transformation* of type $F \leftarrow G$ if, for all $R : A \leftarrow B$,

$$FR \circ \theta_B \;=\; \theta_A \circ GR \ .$$

Whilst, in general, equality does *not* hold, it is the case that equality does hold for all natural transformations when the relation R is a total function. That is, if θ is a natural transformation of type $F \hookleftarrow G$, it is the case that, for all *functions* f of type $A \leftarrow B$,

$$Ff \circ \theta_B \;=\; \theta_A \circ Gf \ . \tag{4}$$

(See [9, lemma 2.70].) Our only use of this property is when f is $!_B$. Specifically, since $!_B$ is a function of type $1 \leftarrow B$,

$$F!_B \circ \theta_B \;=\; \theta_1 \circ G!_B \ . \tag{5}$$

2.5 Membership and Fans

Informally, a natural transformation is a transformation of one type of structure to another type of structure that rearranges the stored information in some way but does no actual computations on the stored information. Hoogendijk and De Moor have made this precise [11]. Their argument, briefly summarised here, is based on the thesis that a datatype (called a "collection type" in their paper) is a relator with a membership relation.

Suppose F is an endorelator[2]. The interpretation of FR is a relation between F structures of the same shape such that corresponding values stored in the two structures are related by R. Suppose A is an object and suppose $X \subseteq \mathsf{id}_A$. So X is a coreflexive; in effect, X selects a subset of A, those values standing in the relation X to themselves. By the same token, FX is the coreflexive that selects all F structures in which all the stored values are members of the subset selected by X. This informal reasoning is the basis of the definition of a membership relation for the datatype F.

The precise specification of membership for F is a collection of relations $\mathsf{mem}.F$ (indexed by objects of the source allegory of F) such that $(\mathsf{mem}.F)_A : A \leftarrow F.A$ and such that FX is the largest subset Y of id_{FA} whose "members" are elements of the set X. Formally, for all A, mem_A is required to satisfy the property:

$$Y \subseteq FX \quad\equiv\quad ((\mathsf{mem}.F)_A \circ Y)< \;\subseteq\; X \tag{6}$$

for all X and Y such that $X \subseteq \mathsf{id}_A$ and $Y \subseteq \mathsf{id}_{FA}$.

A property equivalent to (6) (in fact, the property used in [11] to define membership) is the following: for all R of type $A \leftarrow B$,

$$FR \circ (\mathsf{mem}.F)_B \backslash \mathsf{id}_B \;=\; (\mathsf{mem}.F)_A \backslash R \ . \tag{7}$$

The family of relations $\mathsf{mem}.F \backslash \mathsf{id}$ is called the *fan* of relator F and is denoted below by $\mathsf{fan}.F$. The interpretation of $\mathsf{fan}.F$ is a relation that, given a seed value x, non-deterministically constructs an F structure in which the only stored value is x. For example, given the input value x, $\mathsf{fan.List}$ constructs a list of x s of arbitrary length. (The name "fan" is chosen to suggest an analogy with "fanning" a deck of cards, or the fans with multiple blades used as cooling devices.)

A simple consequence of (7) is that, for all relations R and relators F,

$$F\perp\!\!\!\perp \circ R \;\subseteq\; F\perp\!\!\!\perp \circ \mathsf{fan}.F \ . \tag{8}$$

This fact, which we exploit later, gives an opportunity to illustrate our style of calculation. First note that we have omitted the type information on R and on $\perp\!\!\!\perp$. Formally, the property is that, for all objects A, B and C, and all relations R of type $FA \leftarrow B$,

$$F\perp\!\!\!\perp_{C,A} \circ R \;\subseteq\; F\perp\!\!\!\perp_{C,B} \circ (\mathsf{fan}.F)_B \ . \tag{9}$$

Sometimes the type information is necessary; here it is not. We discuss this further later. The calculation itself then proceeds as follows.

$$
\begin{array}{cl}
 & F\perp\!\!\!\perp \circ R \;\subseteq\; F\perp\!\!\!\perp \circ \mathsf{fan}.F \\
= & \{ \quad (7) \quad \} \\
 & F\perp\!\!\!\perp \circ R \;\subseteq\; \mathsf{mem}.F \backslash \perp\!\!\!\perp \\
= & \{ \quad \text{Galois connection defining division} \quad \} \\
 & \mathsf{mem}.F \circ F\perp\!\!\!\perp \circ R \;\subseteq\; \perp\!\!\!\perp \\
\Leftarrow & \{ \quad \mathsf{mem}.F : \mathsf{Id} \hookleftarrow F\,, \\
 & \quad\; \text{monotonicity of composition} \quad \} \\
 & \perp\!\!\!\perp \circ \mathsf{mem}.F \circ R \;\subseteq\; \perp\!\!\!\perp \\
= & \{ \quad \perp\!\!\!\perp \text{ is zero of composition} \quad \} \\
 & \mathsf{true} \ .
\end{array}
$$

The style, we hope, is self-explanatory. (Note that we use " $=$ " for boolean equality where others might use " $\equiv$ " or even " $\Leftrightarrow$ ".) In

[2] Endorelators have equal source and target allegories. We consider only endorelators in this paper. Hoogendijk [9] shows how to extend the theory to non-endo relators.

this example, we have spelt out all steps in detail. Later we combine simple steps into one step. Hopefully, the reader will be able to supply the missing type information for themself.

A second consequence of (7), proved by Hoogendijk [9, 4.42], is the following:

$$\mathsf{fan}.F \circ (\mathsf{fan}.F)^{\cup} \;\cap\; F\top\!\!\top \quad\subseteq\quad F\mathsf{id}\ . \tag{10}$$

This property expresses the fact that any two F structures that are constructed from the same seed by $\mathsf{fan}.F$ (i.e. are related by $\mathsf{fan}.F \circ (\mathsf{fan}.F)^{\cup}$) and have the same shape (i.e. are related by $F\top\!\!\top$) are equal (i.e. are related by $F\mathsf{id}$, which is the equality relation on F structures).

A third consequence of (7), again proved by Hoogendijk [9, 4.39], is that

$$F\top\!\!\top_{B,A} \circ (\mathsf{fan}.F)_B \;=\; \top\!\!\top_{FB,A}\ .$$

It follows that $(\mathsf{fan}.F)_1 = \top\!\!\top_{F1,1}$ and, hence,

$$((\mathsf{fan}.F)_1){<} \;=\; \mathsf{id}_{F1}\ . \tag{11}$$

The interpretation of datatype $F1$ is the type of F shapes. For example, $\mathsf{List1}$ is isomorphic with the set of natural numbers — the "shape" of a list is its length. The interpretation of (11) is thus that $(\mathsf{fan}.F)_1$ constructs all possible F shapes. (This is one place where the type information is essential! It is not the case that for arbitrary A, $((\mathsf{fan}.F)_A){<} = \mathsf{id}_{FA}$.)

Membership and fans are both natural transformations. Specifically, $\mathsf{mem}.F$ is a natural transformation of type $\mathsf{Id} \hookleftarrow F$ (where Id is the identity relator), and $\mathsf{fan}.F$ is a natural transformation of type $F \hookleftarrow \mathsf{Id}$. More importantly, they are both the largest natural transformations of their type; moreover $\mathsf{mem}.F \backslash \mathsf{mem}.G$ is the largest natural transformation of type $F \hookleftarrow G$ [11]. Formally, if θ is a natural transformation of type $F \hookleftarrow G$,

$$\mathsf{mem}.F \circ \theta \;\subseteq\; \mathsf{mem}.G\ . \tag{12}$$

In words, the members of the F structure constructed by θ are members of the input G structure. The interpretation of this remarkable result is that a natural transformation of type $F \hookleftarrow G$ constructs an F structure from a G structure whereby the stored values in the F structure are rearrangements of the values in the G structure; omission and/or duplication may occur but no creation of new values. This confirms formally the functional programmer's informal understanding of the notion of a polymorphic function. (Note how concise and simple the formulation of the property is!)

2.6 Empty and Non-Empty Structures

A complication in some of our calculations is that we are obliged to perform a case analysis on "empty" and "non-empty" F structures. (For example, an "empty" list is a list of length 0.) Informally, the coreflexive $F\bot\!\!\bot$ is an "empty" F structure. Formally, $(\mathsf{mem}.F){>}$ is the set of non-empty F structures; conversely, $F\bot\!\!\bot$ is the set of F structures that have no members, as we show below.

LEMMA 13. $F\bot\!\!\bot \;=\; (\mathsf{mem}.F){>\!\!\bullet}$.

Proof By mutual inclusion. First,

$$
\begin{array}{ll}
 & F\bot\!\!\bot \;\subseteq\; (\mathsf{mem}.F){>\!\!\bullet} \\
= & \{ \quad \text{definition of negated domain } ({>\!\!\bullet}) \quad \} \\
 & \mathsf{mem}.F \circ F\bot\!\!\bot \;\subseteq\; \bot\!\!\bot \\
\Leftarrow & \{ \quad \mathsf{mem}.F : \mathsf{Id} \hookleftarrow F \ \text{(and transitivity of } \subseteq) \quad \} \\
 & \bot\!\!\bot \circ \mathsf{mem}.F \;\subseteq\; \bot\!\!\bot \\
= & \{ \quad \bot\!\!\bot \text{ is the zero of composition} \quad \} \\
 & \mathsf{true}\ .
\end{array}
$$

Second,

$$
\begin{array}{ll}
 & F\bot\!\!\bot \;\supseteq\; (\mathsf{mem}.F){>\!\!\bullet} \\
= & \{ \quad (6) \quad \} \\
 & \bot\!\!\bot \;\supseteq\; (\mathsf{mem}.F \circ (\mathsf{mem}.F){>\!\!\bullet}){<} \\
= & \{ \quad \text{by definition of } {>\!\!\bullet}, \\
 & \quad\ \ \mathsf{mem}.F \circ (\mathsf{mem}.F){>\!\!\bullet} = \bot\!\!\bot \quad \} \\
 & \bot\!\!\bot \supseteq \bot\!\!\bot{<} \\
= & \{ \quad \text{domains} \quad \} \\
 & \mathsf{true}\ .
\end{array}
$$

□

The contextual information that we are considering a non-empty F structure is expressed by precomposing expressions with the term $(\mathsf{mem}.F){>}$. The following lemma allows us to move the contextual information around in the calculation.

LEMMA 14. If θ is a natural transformation of type $F \hookleftarrow G$ then, for all types A and B and all S of type $GA \leftarrow GB$,

$$\theta{>} \circ S \;=\; \theta{>} \circ S \circ \theta{>} \;=\; S \circ \theta{>} \quad\Leftarrow\quad S \subseteq G\top\!\!\top\ .$$

Proof We prove just the second equality. The first is slightly easier to prove because, unlike the second, no converses are involved.

$$
\begin{array}{ll}
 & \theta{>} \circ S \circ \theta{>} \;=\; S \circ \theta{>} \\
= & \{ \quad \text{domains} \quad \} \\
 & (S \circ \theta{>}){<} \;\subseteq\; \theta{>} \\
= & \{ \quad \text{domains} \quad \} \\
 & (S \circ \theta^{\cup}){<} \;\subseteq\; \theta{>} \\
\Leftarrow & \{ \quad \text{assumption: } S \subseteq G\top\!\!\top \text{, monotonicity} \quad \} \\
 & (G\top\!\!\top \circ \theta^{\cup}){<} \;\subseteq\; \theta{>} \\
= & \{ \quad \text{properties of } {}^{\cup} \quad \} \\
 & (\theta \circ G\top\!\!\top){>} \;\subseteq\; \theta{>} \\
\Leftarrow & \{ \quad \theta : F \hookleftarrow G \text{, monotonicity of } {>} \quad \} \\
 & (F\top\!\!\top \circ \theta){>} \;\subseteq\; \theta{>} \\
= & \{ \quad \text{domains} \quad \} \\
 & \mathsf{true}\ .
\end{array}
$$

□

COROLLARY 15. If θ is a natural transformation of type $F \hookleftarrow G$ then $\theta{>}$ is a proper natural transformation of type $G \leftarrow G$.

Proof Immediate from lemma 14 and the definition of a proper natural transformation.

□

THEOREM 16. $F\bot\!\!\bot$ and $(\mathsf{mem}.F){>}$ are both proper natural transformations of type $F \leftarrow F$. (To be precise, by $F\bot\!\!\bot$ we mean the mapping from object A to $F\bot\!\!\bot_{A,A}$.)

Proof That $F\bot\!\!\bot$ is a proper natural transformation of type $F \leftarrow F$ is immediate from the fact that $\bot\!\!\bot$ is a proper natural transformation of type $\mathsf{Id} \leftarrow \mathsf{Id}$ and the typing rules for natural transformations.

That $(\mathsf{mem}.F){>}$ is a proper natural transformation of type $F{\leftarrow}F$ is an instance of corollary 15.
□

2.7 Fans Make Copies

Fans are natural transformations. Specifically, $\mathsf{fan}.F : F \hookleftarrow \mathsf{Id}$. Recalling (4), this has the consequence that for all relations R that are both simple and total (i.e. functions),

$$\mathsf{fan}.F \circ R \;=\; FR \circ \mathsf{fan}.F \;\; .$$

When a relation R is simple but not total, it is not the case that an equality holds. For example, when R is the empty relation, the left domain of $FR \circ \mathsf{fan}.F$ is $F{\perp\!\!\perp}$; but $\mathsf{fan}.F \circ R$ is the empty relation. The equality does hold when R is simple and we restrict the left domain to $(\mathsf{mem}.F){>}$. Informally, $\mathsf{fan}.F$ is a non-deterministic mapping of a value to an F structure in which all stored values in the F structure are copies of the given value; since simple relations are deterministic, it makes no difference whether the relation R is applied to the given value before or after the copying takes place. We prove this in theorem 18; first we need a preliminary lemma.

LEMMA 17. For all R and S,

$$S{>} \circ S\backslash R \;\;\subseteq\;\; S\backslash R \circ R^{\cup} \circ R \;\; .$$

Proof

$$
\begin{array}{ll}
 & S{>} \circ S\backslash R \\
\subseteq & \{ \quad S{>} \;=\; \mathsf{id} \cap S^{\cup} \circ S \;, \;\; \text{monotonicity} \quad \} \\
 & S\backslash R \;\cap\; S^{\cup} \circ S \circ S\backslash R \\
\subseteq & \{ \quad \text{cancellation of factors, monotonicity} \quad \} \\
 & S\backslash R \;\cap\; S^{\cup} \circ R \\
\subseteq & \{ \quad \text{modular law, monotonicity} \quad \} \\
 & S\backslash R \circ R^{\cup} \circ R \;\; .
\end{array}
$$

□

THEOREM 18. For all simple R,

$$(\mathsf{mem}.F){>} \circ FR \circ \mathsf{fan}.F \;\;=\;\; (\mathsf{mem}.F){>} \circ \mathsf{fan}.F \circ R \;\; .$$

Proof By mutual inclusion. First,

$$
\begin{array}{ll}
 & (\mathsf{mem}.F){>} \circ FR \circ \mathsf{fan}.F \\
\supseteq & \{ \quad \mathsf{fan}.F : F \hookleftarrow \mathsf{Id} \;, \\
 & \quad\;\; \text{monotonicity of composition} \quad \} \\
 & (\mathsf{mem}.F){>} \circ \mathsf{fan}.F \circ R \;\; .
\end{array}
$$

Second,

$$
\begin{array}{ll}
 & (\mathsf{mem}.F){>} \circ FR \circ \mathsf{fan}.F \\
 & \subseteq \;\; (\mathsf{mem}.F){>} \circ \mathsf{fan}.F \circ R \\
= & \{ \quad (\mathsf{mem}.F){>} \text{ is a coreflexive} \quad \} \\
 & (\mathsf{mem}.F){>} \circ FR \circ \mathsf{fan}.F \;\;\subseteq\;\; \mathsf{fan}.F \circ R \\
= & \{ \quad FR \circ \mathsf{fan}.F \;\;=\;\; \mathsf{mem}.F\backslash R \quad \} \\
 & (\mathsf{mem}.F){>} \circ \mathsf{mem}.F\backslash R \;\;\subseteq\;\; \mathsf{fan}.F \circ R \\
\Leftarrow & \{ \quad \text{lemma 17, } S := \mathsf{mem}.F \quad \} \\
 & \mathsf{mem}.F\backslash R \circ R^{\cup} \circ R \;\;\subseteq\;\; \mathsf{mem}.F\backslash \mathsf{id} \circ R \\
\Leftarrow & \{ \quad \text{monotonicity and factors} \quad \} \\
 & \mathsf{mem}.F \circ \mathsf{mem}.F\backslash R \circ R^{\cup} \;\;\subseteq\;\; \mathsf{id} \\
\Leftarrow & \{ \quad \text{cancellation of factors} \quad \} \\
 & R \circ R^{\cup} \subseteq \mathsf{id} \\
= & \{ \quad \text{definition} \quad \} \\
 & \mathsf{simple}.R \;\; .
\end{array}
$$

□

3. The Fan Club

3.1 Definitions and Naming

Given datatypes F, G and H, there are six different ways of composing their fans to form a natural transformation of type $FGH \hookleftarrow \mathsf{Id}$ These are given below. Simultaneously, we name them using a combination of the letters “f”, “g” and “h” to indicate the order of composition.

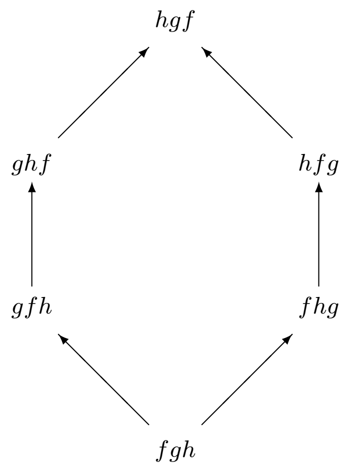
$$
\begin{array}{lcl}
fgh & = & (\mathsf{fan}.F)GH \circ (\mathsf{fan}.G)H \circ \mathsf{fan}.H \\
gfh & = & F(\mathsf{fan}.G)H \circ (\mathsf{fan}.F)H \circ \mathsf{fan}.H \\
fhg & = & (\mathsf{fan}.F)GH \circ G(\mathsf{fan}.H) \circ \mathsf{fan}.G \\
hfg & = & FG(\mathsf{fan}.H) \circ (\mathsf{fan}.F)G \circ \mathsf{fan}.G \\
ghf & = & F(\mathsf{fan}.G)H \circ F(\mathsf{fan}.H) \circ \mathsf{fan}.F \\
hgf & = & FG(\mathsf{fan}.H) \circ F(\mathsf{fan}.G) \circ \mathsf{fan}.F
\end{array}
$$

We leave the reader to check the types. For example, considering gfh, $F(\mathsf{fan}.G)H$ has type $FGH \hookleftarrow FH$, $(\mathsf{fan}.F)H$ has type $FH \hookleftarrow H$ and $\mathsf{fan}.H$ has type $H \hookleftarrow \mathsf{Id}$. Hence, gfh has type $FGH \hookleftarrow \mathsf{Id}$ (using the rule that, if η has type $F \hookleftarrow G$ and τ has type $G \hookleftarrow H$, $\eta \circ \tau$ hs type $F \hookleftarrow H$).

These are the members of our fanclub. In the remaining sections, we investigate the relation between them. We begin in section 3.2 by observing how they may be partially ordered; this ordering then forms the basis for a discussion of some concrete examples. Next, in section 3.3, we give a diagrammatic summary of the relationship between the semilattice formed by the members of the club under set intersection. The precise relationships are stated and proved in section 5. Following this, we return to our introductory problem in section 4.2; as we show, the problem is indeed a straightforward corollary of the properties of the fanclub.

3.2 Inclusions and Examples

Natural transformations of the same type can be ordered by inclusion everywhere. That is, transformation η is *included everywhere* in transformation τ if, for all A, $\eta_A \subseteq \tau_A$. The members of the fanclub are ordered in this way as shown below. Each arrow indicates an inclusion everywhere (i.e. for all instances of the transformations). For example, the arrow from gfh to ghf asserts that, everywhere, $gfh \subseteq ghf$.



The inclusions follow from the naturality of fans and the monotonicity of composition. Here, for example, is the proof of the inclusion $gfh \subseteq ghf$.

$$
\begin{array}{ll}
 & gfh \\
= & \{ \quad \text{definition} \quad \} \\
 & F(\mathsf{fan}.G)H \circ (\mathsf{fan}.F)H \circ \mathsf{fan}.H \\
\subseteq & \{ \quad \mathsf{fan}.F : F \hookleftarrow \mathsf{Id} \text{ , hence} \\
 & \quad (\mathsf{fan}.F)H \circ \mathsf{fan}.H \subseteq F(\mathsf{fan}.H) \circ \mathsf{fan}.F \quad \} \\
 & F(\mathsf{fan}.G)H \circ F(\mathsf{fan}.H) \circ \mathsf{fan}.F \\
= & \{ \quad \text{definition} \quad \} \\
 & ghf \ .
\end{array}
$$

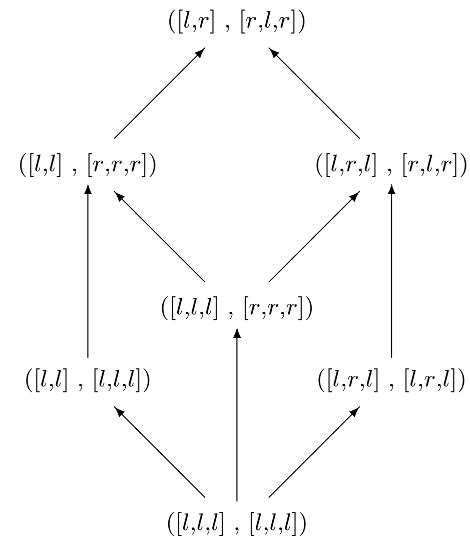
The crucial middle step of the above calculation can also be justified by appeal to a stronger property of fans which we exploit later. In the above example, the component " hf " in ghf is $F(\mathsf{fan}.H) \circ \mathsf{fan}.F$, which equals $\mathsf{fan}.FH$, the largest natural transformation of type $FH \hookleftarrow \mathsf{Id}$. The corresponding component " fh " in gfh is also a natural transformation of type $FH \hookleftarrow \mathsf{Id}$, and thus included in $\mathsf{fan}.FH$. Monotonicity of relators and composition then completes the proof. The same argument can be applied to all six edges in the inclusion diagram because each edge corresponds to switching around two of the letters in the names — for example, " h " and " f " are switched in the arrow from gfh to ghf — with the upper component being a fan of the composition of two relators and thus the largest natural transformation of its type.

Formally, we have, for all relators F and G ,

$$\mathsf{fan}.FG \quad = \quad F(\mathsf{fan}.G) \circ \mathsf{fan}.F \ . \tag{19}$$

The simple proof by mutual inclusion uses (7) and the naturality properties of $\mathsf{mem}.F$.

The relation between the fans can be illustrated by considering specific relators F , G and H . In the diagram below, F is the doubling relator, Δ , defined by $\Delta X = X \times X$; the fan of Δ relates the pair (x , x) to x (for arbitrary x , irrespective of its type). Also, the relator G is List ; the fan of List relates a list (of arbitrary length) to a value x equivales every element of the list equals x . Finally, the relator H is the sum relator, ∇ , defined by $\nabla X = X + X$; the fan of ∇ relates the pairs (l, x) and (r, x) to x . (" l " and " r " are so-called "tags", normally denoted by inl and inr ; we use l and r here for brevity.)



Each node of the diagram is an example of a pair of lists of sums related by a combination of fans to a given seed; the value of the seed is irrelevant and has been omitted, allowing us to abbreviate (l, x) to l and (r, x) to r . The central node has been added; the reason for its addition is discussed shortly.

When studying the diagram, it is important to bear in mind that the shape of a list is its length, and the shape of a value in a sum type is its tag, " l " or " r ". (All pairs have the same shape; in general, the shape of the F structure is not relevant.)

The top and bottom nodes are the easiest to describe. The top node is an arbitrary pair of lists of sums, the bottom node is a pair of lists of the same length such that all the elements of all the lists are equal.

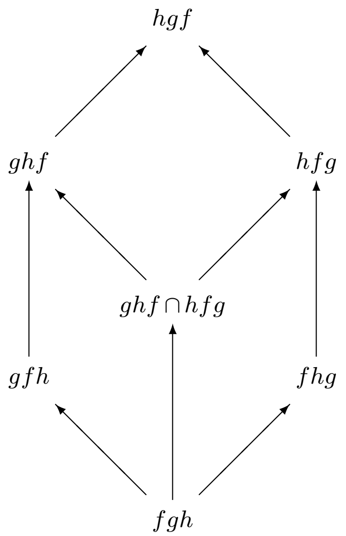
The top-left node is a pair of lists, each of arbitrary length, such that, for each list, its elements are equal. The bottom-left node has the further restriction that all the elements in all the lists are equal.

The top-right node is a pair of lists of equal length. The bottom-right node has the further restriction that the two lists are equal.

The central node has been added to the diagram. This node gives an example of a structure related to the seed by $ghf \cap hfg$. It is a pair of lists of equal length such that, for each list, its elements are all equal.

3.3 The Goal

Our goal is to detail how the members of the fanclub are related by set intersection. The diagram below summarises the conclusions. Specifically, the intersection of two members of the fanclub is given by the highest node in the diagram that is "below" (i.e. included in) both members. For example, $gfh \cap hfg$ equals fgh . In words, an FGH structure such that all the H structures have the same shape and all the G structures have the same shape has the property that all the GH structures have the same shape.



Note that the diagram does not show the relationship between members of the fanclub under set union. In this context, set union is less interesting than set intersection because the former is just about a disjunction of shape properties whereas the latter is about a conjunction of shape properties.

Note also that the example of pairs of lists of sums presented above demonstrates that the members of the fanclub are different. Strictly, we should establish this formally (for example, by showing that ($[l,l]$, $[r,r,r]$) cannot be generated by hfg). That exercise we leave to the reader.

4. Shape Selectors

The six members of the fanclub listed in section 3.1 are all natural transformations of type $FGH \hookleftarrow \mathsf{Id}$. Before investigating their properties, we explore the properties of natural tranformations of type $F \hookleftarrow \mathsf{Id}$ for arbitrary F . Among the properties we prove is theorem 24, which asserts that such families of relations are closed under intersection. It follows that all seven elements of the club shown in the figure in section 3.3 are natural transformations of type $FGH \hookleftarrow \mathsf{Id}$.

4.1 Reduction to the Unit Type

Natural transformations of type $F \hookleftarrow \mathsf{Id}$ are subsets of $\mathsf{fan}.F$. Whilst $\mathsf{fan}.F$ can be viewed as a nondeterministic program that constructs F structures of all possible shapes, a typical natural transformation of type $F \hookleftarrow \mathsf{Id}$ constructs only some F structures but in a way that is independent of the type of the argument (i.e. is so-called "natural"). In essence, natural transformations of type $F \hookleftarrow \mathsf{Id}$ select F shapes.

Natural transformations of type $K_1 \hookleftarrow F$ also select F shapes. If φ has type $K_1 \hookleftarrow F$ then, for each object A , φ_A is a relation between 1 and FA ; that is, φ_A selects elements in some subset of FA (specifically, the subset given by $\varphi_A{}^>$). But, naturality of φ means that the selection is made independently of A ; that is, φ selects on the basis of the shape of the structure, not on the stored values.

Formally, there is an order isomorphism between natural transformations of type $F \hookleftarrow \mathsf{Id}$ and natural transformations of type $K_1 \hookleftarrow F$ (theorem 29).

Natural transformations of type $F \hookleftarrow \mathsf{Id}$ are completely characterised by their behaviour at the unit type. This is the essence of corollary 22 below. First, we show how to express θ_A , for natural transformation θ of type $F \hookleftarrow \mathsf{Id}$, in terms of θ_1 . (Note that the type information is included in this calculation because that is what the calculation is about!)

THEOREM 20. If θ has type $F \hookleftarrow \mathsf{Id}$ then, for all A ,

$$\theta_A \;=\; (F!_A{}^{\cup} \circ \theta_1 \circ !_A) \cap (\mathsf{fan}.F)_A \;\; .$$

Proof By mutual inclusion.

$$
\begin{array}{ll}
& (F!_A{}^{\cup} \circ \theta_1 \circ !_A) \cap (\mathsf{fan}.F)_A \\
= & \{ \quad \theta \text{ has type } F \hookleftarrow \mathsf{Id} \text{ , } !_A \text{ is a function} \quad \} \\
& (F!_A{}^{\cup} \circ F!_A \circ \theta_A) \cap (\mathsf{fan}.F)_A \\
\subseteq & \{ \quad \text{modular law} \quad \} \\
& ((F!_A{}^{\cup} \circ F!_A) \cap ((\mathsf{fan}.F)_A \circ \theta_A^{\cup})) \circ \theta_A \\
\subseteq & \{ \quad F!_A{}^{\cup} \circ F!_A \;\subseteq\; F\top\!\!\top_{A,A} \text{ ,} \\
& \quad\; \theta \subseteq \mathsf{fan}.F \text{ (because } \mathsf{fan}.F \text{ is the largest} \\
& \quad\; \text{natural transformation of its type)} \quad \} \\
& (F\top\!\!\top_{A,A} \cap ((\mathsf{fan}.F)_A \circ (\mathsf{fan}.F)_A^{\cup})) \circ \theta_A \\
\subseteq & \{ \quad (10) \quad \} \\
& \theta_A \\
\subseteq & \{ \quad !_A \text{ is total; i.e. } \mathsf{id}_A \;\subseteq\; !_A{}^{\cup} \circ !_A \text{ ,} \\
& \quad\; \theta \subseteq \mathsf{fan}.F \quad \} \\
& (F(!_A{}^{\cup} \circ !_A) \circ \theta_A) \cap (\mathsf{fan}.F)_A \\
= & \{ \quad \text{distributivity} \quad \} \\
& (F!_A{}^{\cup} \circ F!_A \circ \theta_A) \cap (\mathsf{fan}.F)_A \\
= & \{ \quad \theta \text{ has type } F \hookleftarrow \mathsf{Id} \text{ , } !_A \text{ is a function} \quad \} \\
& (F!_A{}^{\cup} \circ \theta_1 \circ !_A) \cap (\mathsf{fan}.F)_A \;\; .
\end{array}
$$

□

Theorem 20 expresses θ_A , for arbitrary A , in terms of θ_1 . In turn, $\theta_1 \circ !_A$ can be expressed in terms of the left domain of θ_1 . Specifically, since $!_A = \top\!\!\top_{1,A}$ and $\theta_1 \circ \top\!\!\top_{1,A} \;=\; \theta_1{}^< \circ \top\!\!\top_{F1,A}$,

$$\theta_1 \circ !_A \;=\; \theta_1{}^< \circ \top\!\!\top_{F1,A} \;\; . \qquad (21)$$

It follows that every natural transformation of type $F \hookleftarrow \mathsf{Id}$ is completely determined by its left domain at the unit type. More specifically:

COROLLARY 22. If θ and φ both have type $F \hookleftarrow \mathsf{Id}$ then,

$$(\theta \subseteq \varphi) \;=\; (\theta_1 \subseteq \varphi_1) \;=\; (\theta_1{}^< \subseteq \varphi_1{}^<) \;=\; (\theta^< \subseteq \varphi^<) \;\; .$$

Proof Straightforward ping-pong proof using theorem 20, (21) and the monotonicity of the left-domain operator.

□

COROLLARY 23. If θ has type $F \hookleftarrow \mathsf{Id}$ then, for all A ,

$$\theta_A{}^< \;=\; (F!_A{}^{\cup} \circ \theta_1)^< \circ (\mathsf{fan}.F)_A{}^< \;\; .$$

Proof

$$
\begin{array}{ll}
& \theta_A{}^< \\
= & \{ \quad \text{theorem 20} \quad \} \\
& ((F!_A{}^{\cup} \circ \theta_1 \circ !_A) \cap (\mathsf{fan}.F)_A)^< \\
= & \{ \quad \text{for all } R \text{, } S \text{ and } T \text{,} \\
& \quad\; ((R \circ S) \cap T)^< \;=\; (R \cap (T \circ S^{\cup}))^< \quad \}
\end{array}
$$

$$((F!_A{}^{\cup} \circ \theta_1) \;\cap\; (\mathsf{fan}.F)_A \circ !_A{}^{\cup})<$$

$$= \quad \{ \quad F!_A{}^{\cup} \circ \theta_1 \text{ and } (\mathsf{fan}.F)_A \circ !_A{}^{\cup} \text{ have source } 1 \text{ , distributivity } \}$$

$$(F!_A{}^{\cup} \circ \theta_1)< \;\cap\; ((\mathsf{fan}.F)_A \circ !_A{}^{\cup})<$$

$$= \quad \{ \quad \text{coreflexives, } (!_A{}^{\cup})< = \mathsf{id}_A \quad \}$$

$$(F!_A{}^{\cup} \circ \theta_1)< \circ (\mathsf{fan}.F)_A< \;.$$

□

THEOREM 24. If θ and φ both have type $F \hookleftarrow \mathsf{Id}$ then $\theta \cap \varphi$ is a natural transformation of type $F \hookleftarrow \mathsf{Id}$.

Proof For all R of type $A \leftarrow B$, we have:

$$(\theta_A \cap \varphi_A) \circ R$$

$$= \quad \{ \quad \text{theorem 20, properties of intersection} \quad \}$$

$$((F!_A{}^{\cup} \circ \theta_1 \circ !_A) \;\cap\; (F!_A{}^{\cup} \circ \varphi_1 \circ !_A) \;\cap\; (\mathsf{fan}.F)_A) \circ R$$

$$\subseteq \quad \{ \quad \text{monotonicity of composition} \quad \}$$

$$(F!_A{}^{\cup} \circ \theta_1 \circ !_A \circ R)$$

$$\cap \quad (F!_A{}^{\cup} \circ \varphi_1 \circ !_A \circ R)$$

$$\cap \quad ((\mathsf{fan}.F)_A \circ R)$$

$$\subseteq \quad \{ \quad !_A \circ R \subseteq !_B \text{ , } \mathsf{fan}.F \text{ has type } F \hookleftarrow \mathsf{Id} \quad \}$$

$$(F!_A{}^{\cup} \circ \theta_1 \circ !_B)$$

$$\cap \quad (F!_A{}^{\cup} \circ \varphi_1 \circ !_B)$$

$$\cap \quad (FR \circ (\mathsf{fan}.F)_B)$$

$$\subseteq \quad \{ \quad \text{modular law, monotonicity of composition, } !_A \circ R \subseteq !_B \quad \}$$

$$FR \circ ((F!_B{}^{\cup} \circ \theta_1 \circ !_B) \;\cap\; (F!_B{}^{\cup} \circ \varphi_1 \circ !_B) \;\cap\; (\mathsf{fan}.F)_B)$$

$$= \quad \{ \quad \text{theorem 20, properties of intersection} \quad \}$$

$$FR \circ (\theta_B \cap \varphi_B) \;.$$

□

DEFINITION 25 (Shape-selector Isomorphisms). With θ ranging over natural transformations of type $F \hookleftarrow \mathsf{Id}$, and φ ranging over natural transformations of type $K_1 \hookleftarrow F$, we define the mappings $\theta 2\varphi$ and $\varphi 2\theta$ by:

$$(\theta 2\varphi.\theta)_A \;=\; \theta_1{}^{\cup} \circ F!_A$$

and

$$(\varphi 2\theta.\varphi)_A \;=\; (F!_A{}^{\cup} \circ \varphi_1{}^{\cup} \circ !_A) \cap (\mathsf{fan}.F)_A \;.$$

□

LEMMA 26. (a) If S is a relation of type $1 \leftarrow F1$ then $S \circ F!$ is a natural transformation of type $K_1 \hookleftarrow F$. (b) If S is a relation of type $F1 \leftarrow 1$ then $((F!)^{\cup} \circ S \circ !) \cap \mathsf{fan}.F$ is a natural transformation of type $F \hookleftarrow \mathsf{Id}$.

(To be completely correct, we should write K_S and not S ; K_S is the natural transformation of type $K_1 \hookleftarrow K_{F1}$ such that, for all A , $(K_S)_A = S$.)

Proof (a) That $S \circ F!$ has type $K_1 \hookleftarrow F$ is a straightforward application of the typing rules for natural transformations since $!$ has type $K_1 \hookleftarrow \mathsf{Id}$. (b) For any R of type $A \leftarrow B$, we have:

$$((F!_A{}^{\cup} \circ S \circ !_A) \cap (\mathsf{fan}.F)_A) \circ R$$

$$\subseteq \quad \{ \quad \text{monotonicity} \quad \}$$

$$F!_A{}^{\cup} \circ S \circ !_A \circ R \quad \cap \quad (\mathsf{fan}.F)_A \circ R$$

$$\subseteq \quad \{ \quad !_A \circ R \subseteq !_B \text{ , } \mathsf{fan}.F \text{ has type } F \hookleftarrow \mathsf{Id} \quad \}$$

$$F!_A{}^{\cup} \circ S \circ !_B \quad \cap \quad FR \circ (\mathsf{fan}.F)_B$$

$$\subseteq \quad \{ \quad \text{modular law} \quad \}$$

$$FR \circ (FR^{\cup} \circ F!_A{}^{\cup} \circ S \circ !_B \quad \cap \quad (\mathsf{fan}.F)_B)$$

$$\subseteq \quad \{ \quad !_A \circ R \subseteq !_B \quad \}$$

$$FR \circ (F!_B{}^{\cup} \circ S \circ !_B \quad \cap \quad (\mathsf{fan}.F)_B) \;.$$

The lemma follows by definition.

□

LEMMA 27. If φ has type $K_1 \hookleftarrow F$ then, for all objects A ,

$$\varphi_A \;=\; \varphi_1 \circ F!_A \;.$$

Proof Instantiate (5) and use the fact that $\mathsf{id}_1 = !_1$.

□

COROLLARY 28. If θ and φ both have type $K_1 \hookleftarrow F$ then,

$$(\theta \subseteq \varphi) \;=\; (\theta_1 \subseteq \varphi_1) \;=\; (\theta_1> \subseteq \varphi_1>) \;=\; (\theta> \subseteq \varphi>) \;.$$

Proof Straightforward ping-pong proof using lemma 27.

□

THEOREM 29. The mappings defined in definition 25 form an order isomorphism between natural transformations of type $F \hookleftarrow \mathsf{Id}$ and natural transformations of type $K_1 \hookleftarrow F$.

Proof By lemma 26(a), $\theta 2\varphi$ maps a natural transformation of type $F \hookleftarrow \mathsf{Id}$ into a natural transformation of type $K_1 \hookleftarrow F$. Similarly, lemma 26(b) proves that $\varphi 2\theta$ does the reverse.

We prove that $\theta 2\varphi$ is an order isomorphism as follows. Suppose α and β are both natural transformations of type $F \hookleftarrow \mathsf{Id}$. Then

$$\theta 2\varphi.\alpha \;\subseteq\; \theta 2\varphi.\beta$$

$$= \quad \{ \quad \text{definition 25, corollary 28} \quad \}$$

$$(\alpha_1{}^{\cup} \circ F!_1)> \;\subseteq\; (\beta_1{}^{\cup} \circ F!_1)>$$

$$= \quad \{ \quad !_1 = \mathsf{id}_1 \text{ , domains} \quad \}$$

$$\alpha_1< \;\subseteq\; \beta_1<$$

$$= \quad \{ \quad \text{corollary 22} \quad \}$$

$$\alpha \;\subseteq\; \beta \;.$$

A similar argument establishes that $\varphi 2\theta$ is also an order isomorphism. Suppose α and β are both natural transformations of type $K_1 \hookleftarrow F$. Then

$$\varphi 2\theta.\alpha \;\subseteq\; \varphi 2\theta.\beta$$

$$= \quad \{ \quad \text{definition 25, corollary 22} \quad \}$$

$$(F!_1{}^{\cup} \circ \alpha_1{}^{\cup} \circ !_1) \cap (\mathsf{fan}.F)_1 \;\subseteq\; (F!_1{}^{\cup} \circ \beta_1{}^{\cup} \circ !_1) \cap (\mathsf{fan}.F)_1$$

$$= \quad \{ \quad (\mathsf{fan}.F)_1 = !_{F1}{}^{\cup} = \top\!\!\top_{F1,1} \quad , \quad !_1 = \mathsf{id}_1 \quad \}$$

$$\alpha_1{}^{\cup} \;\subseteq\; \beta_1{}^{\cup}$$

$$= \quad \{ \quad \text{corollary 28} \quad \}$$

$$\alpha \;\subseteq\; \beta \;.$$

□

4.2 The Transformation Problem

We now return to the problem stated in the introduction. It concerns a natural transformation θ with a particular shape property. The goal is to establish a shape property of θH .

The first property we are given of θ is that it has type $FG \hookleftarrow G$. We are also given that θ returns F structures of G structures of the same shape. Shape properties of natural transformations are determined by instantiating them at the unit type, 1 . In this case, θ_1 has type $FG1 \leftarrow G1$. The object $G1$ represents the set of all G shapes. (For example, $1{\times}1$ is isomorphic to 1 —all pairs have the same shape— , $1{+}1$ is isomorphic to Bool —the shape of an element of a disjoint sum is given by its tag— and $\mathsf{List1}$ is isomorphic to the natural numbers —the shape of a list is its length— .) The set of F structures of G structures of the same shape is represented by $((\mathsf{fan}.F)_{G1})_<$. So we are given that

$$(\theta_1)_< \;\subseteq\; ((\mathsf{fan}.F)_{G1})_< \;.$$

The goal is now to show that, if θH is applied to a G structure of H structures all of the same shape, it returns a result that is an F structure of GH structures all of the same shape. Specifically, we show that

$$(\theta H \circ (\mathsf{fan}.G)H \circ \mathsf{fan}.H)_< \;\subseteq\; ((\mathsf{fan}.F)GH)_< \;.$$

This we prove in corollary 31. First we prove a more general result.

LEMMA 30. If θ has type $FG \hookleftarrow \mathsf{Id}$ then

$$\theta \subseteq (\mathsf{fan}.F)G \circ \mathsf{fan}.G \;\equiv\; (\theta_1)_< \subseteq ((\mathsf{fan}.F)_{G1})_< \;.$$

Proof

$$\begin{array}{ll}
 & \theta \subseteq (\mathsf{fan}.F)G \circ \mathsf{fan}.G \\
= & \{ \quad \text{corollary 22} \quad \} \\
 & (\theta_1)_< \subseteq (((\mathsf{fan}.F)G \circ \mathsf{fan}.G)_1)_< \\
= & \{ \quad ((\mathsf{fan}.G)_1)_< = \mathsf{id}_{G1} \;,\; \text{domains: (2)} \quad \} \\
 & (\theta_1)_< \subseteq ((\mathsf{fan}.F)_{G1})_< \;.
\end{array}$$

□

COROLLARY 31. If θ has type $FG \hookleftarrow G$ and

$$(\theta_1)_< \subseteq ((\mathsf{fan}.F)_{G1})_<$$

then

$$(\theta H \circ (\mathsf{fan}.G)H \circ \mathsf{fan}.H)_< \;\subseteq\; ((\mathsf{fan}.F)GH)_< \;.$$

Proof First note that the premise $(\theta_1)_< \subseteq ((\mathsf{fan}.F)_{G1})_<$ equivales

$$((\theta \circ \mathsf{fan}.G)_1)_< \;\subseteq\; (((\mathsf{fan}.F)G)_1)_< \qquad (32)$$

because $((\mathsf{fan}.G)_1)_< = \mathsf{id}_{G1}$. Property (32) is the premise in lemma 30. Thus:

$$\begin{array}{ll}
 & (\theta H \circ (\mathsf{fan}.G)H \circ \mathsf{fan}.H)_< \\
\subseteq & \{ \quad \text{lemma 30, } \theta := \theta \circ \mathsf{fan}.G \text{, (32),} \\
 & \quad\;\; \text{monotonicity} \quad \} \\
 & ((\mathsf{fan}.F)GH \circ (\mathsf{fan}.G)H \circ \mathsf{fan}.H)_< \\
\subseteq & \{ \quad \text{domains: (2)} \quad \} \\
 & ((\mathsf{fan}.F)GH)_< \;.
\end{array}$$

□

5. Meeting the Fanclub

In this section, we relate the intersection ("meet") of pairs of members of the fanclub to the other members of the club. Note that, by theorem 24, all "meets" of fans in the club are natural transformations of type $FGH \hookleftarrow \mathsf{Id}$; this allows us to exploit corollary 22 as the basic tool for establishing inclusions between the members.

THEOREM 33.

$$gfh \cap hfg \;=\; fgh \;.$$

Proof First:

$$\begin{array}{ll}
 & gfh \cap hfg \;=\; fgh \\
= & \{ \quad \text{theorem 24 and corollary 22} \quad \} \\
 & (gfh_1 \cap hfg_1)_< \;=\; (fgh_1)_< \;.
\end{array}$$

Now we continue with the left side of the inclusion.

$$\begin{array}{ll}
 & (gfh_1 \cap hfg_1)_< \\
= & \{ \quad gfh_1 \text{ and } hfg_1 \text{ are left conditions,} \\
 & \quad\;\; \text{distributivity} \quad \} \\
 & (gfh_1)_< \circ (hfg_1)_< \\
= & \{ \quad \text{definition of } gfh \text{, } ((\mathsf{fan}.H)_1)_< = \mathsf{id}_{H1} \quad \} \\
 & ((\mathsf{fan}.FG)_{H1})_< \circ (hfg_1)_< \\
= & \{ \quad \text{definition of } hfg \text{, } (\mathsf{fan}.H)_1 = \overset{\cup}{!}_{H1} \quad \} \\
 & ((\mathsf{fan}.FG)_{H1})_< \circ (FG\overset{\cup}{!}_{H1} \circ ((\mathsf{fan}.F)G)_1 \circ (\mathsf{fan}.G)_1)_< \\
= & \{ \quad \text{corollary 23,} \\
 & \quad\;\; \theta{,}F{,}A := (\mathsf{fan}.F)G \circ \mathsf{fan}.G \,,\, FG \,,\, H1 \quad \} \\
 & (((\mathsf{fan}.F)G \circ \mathsf{fan}.G)_{H1})_< \\
= & \{ \quad ((\mathsf{fan}.H)_1)_< = \mathsf{id}_{H1} \quad \} \\
 & ((\mathsf{fan}.F)GH \circ (\mathsf{fan}.G)H \circ \mathsf{fan}.H)_{1<} \\
= & \{ \quad \text{definition} \quad \} \\
 & (fgh_1)_< \;.
\end{array}$$

□

COROLLARY 34.

$$gfh \cap hfg \;=\; fgh \;=\; gfh \cap fhg \;.$$

Proof Immediate from theorem 33 and the inclusions $fgh \subseteq gfh$ and $fgh \subseteq fhg \subseteq hfg$. (See section 3.2.)
□

At this point, it would be fortunate if we could give a proof similar to the proof of theorem 33 showing that $ghf \cap fhg \;=\; fgh$. However, we have not been able to find such a proof and have to resort to a case analysis on empty and non-empty F structures.

For non-empty structures, we do have the desired equality:

LEMMA 35.

$$((\mathsf{mem}.F)_{GH})_> \circ (ghf \cap fhg) \;=\; ((\mathsf{mem}.F)_{GH})_> \circ fgh \;.$$

Proof Recalling that $(\mathsf{mem}.F)_>$ is a proper natural transformation of type $F \leftarrow F$ (theorem 16), it follows that both the left and right sides of the desired equation are natural transformations of type $FGH \hookleftarrow \mathsf{Id}$. So, as in theorem 33, it suffices to prove that

$$\begin{array}{ll}
 & ((\mathsf{mem}.F)_{GH1})_> \circ ghf_{1<} \circ fhg_{1<} \\
= & ((\mathsf{mem}.F)_{GH1})_> \circ fgh_{1<} \;.
\end{array}$$

Now,

$\quad ghf_1{<} \circ fhg_1{<}$

$=\quad \{ \quad ghf \;=\; F(\mathsf{fan}.G)H \circ \mathsf{fan}.FH$,
$\qquad fhg \;=\; (\mathsf{fan}.F)GH \circ \mathsf{fan}.GH \quad \}$

$\quad ((F(\mathsf{fan}.G)H)_1 \circ (\mathsf{fan}.FH)_1){<}$
$\quad \circ \;\; (((\mathsf{fan}.F)GH)_1 \circ (\mathsf{fan}.GH)_1){<}$

$=\quad \{$ domains: (2),
for all F , $((\mathsf{fan}.F)_1){<} \;=\; \mathsf{id}_{F1}$
with $F := FH$ and $F := GH$ $\}$

$\quad ((F(\mathsf{fan}.G)H)_1){<} \;\circ\; (((\mathsf{fan}.F)GH)_1){<}$.

Also, for all A ,

$\quad (((\mathsf{mem}.F)_{GA}){>} \;\circ\; (F(\mathsf{fan}.G)_A){<} \;\circ\; (\mathsf{fan}.F)_{GA}){<}$

$=\quad \{$ relators and domain operators commute,
domains: (2) $\}$

$\quad (((\mathsf{mem}.F)_{GA}){>} \;\circ\; F((\mathsf{fan}.G_A){<}) \;\circ\; (\mathsf{fan}.F)_{GA}){<}$

$=\quad \{$ theorem 18 with $R := ((\mathsf{fan}.G)_A){<}$,
coreflexives are simple $\}$

$\quad (((\mathsf{mem}.F)_{GA}){>} \;\circ\; (\mathsf{fan}.F)_{GA} \;\circ\; ((\mathsf{fan}.G)_A){<}){<}$

$=\quad \{$ domains: (2) $\}$

$\quad ((\mathsf{mem}.F)_{GA}){>} \;\circ\; ((\mathsf{fan}.F)_{GA} \;\circ\; (\mathsf{fan}.G)_A){<}$.

Thus,

$\quad ((\mathsf{mem}.F)_{GH1}){>} \;\circ\; ghf_1{<} \;\circ\; fhg_1{<}$

$=\quad \{$ first calculation above $\}$

$\quad (((\mathsf{mem}.F)_{GH1}){>} \;\circ\; (F(\mathsf{fan}.G)_{H1}){<} \;\circ\; (\mathsf{fan}.F)_{GH1}){<}$

$=\quad \{$ second calculation with $A := H1$ $\}$

$\quad ((\mathsf{mem}.F)_{GH1}){>} \;\circ\; ((\mathsf{fan}.F)_{GH1} \;\circ\; (\mathsf{fan}.G)_{H1}){<}$

$=\quad \{$ definition of fgh ,
$((\mathsf{fan}.H)_1){<} \;=\; \mathsf{id}_{H1}$ $\}$

$\quad ((\mathsf{mem}.F)_{GH1}){>} \;\circ\; fgh_1{<}$.

□

Now we consider empty F structures.

LEMMA 36.

$$F\perp\!\!\!\perp \circ fhg \;\;=\;\; F\perp\!\!\!\perp \circ fgh \;\;.$$

Proof We have:

$\quad F\perp\!\!\!\perp \circ fhg$

$=\quad \{$ lemma 37 (below) , domains $\}$

$\quad F\perp\!\!\!\perp \circ fhg \circ (\mathsf{fan}.H){>}$

$\subseteq\quad \{$ $(\mathsf{fan}.H){>} \;=\; \mathsf{id} \;\cap\; (\mathsf{fan}.H)^{\cup} \circ \mathsf{fan}.H$,
monotonicity $\}$

$\quad F\perp\!\!\!\perp \circ (fhg \;\cap\; fhg \circ (\mathsf{fan}.H)^{\cup} \circ \mathsf{fan}.H)$

$\subseteq\quad \{$ coreflexive $F\perp\!\!\!\perp$ distributes through
intersection, lemma 38 (below) $\}$

$\quad F\perp\!\!\!\perp \circ (fhg \;\cap\; gfh)$

$=\quad \{$ corollary 34 $\}$

$\quad F\perp\!\!\!\perp \circ fgh$.

The equality follows from the basic inclusion $fgh \subseteq fhg$, monotonicity and anti-symmetry.

□

LEMMA 37.

$$fhg{>} \;\subseteq\; (\mathsf{fan}.H){>}$$

Proof

$\quad fhg{>} \;\subseteq\; (\mathsf{fan}.H){>}$

$\Leftarrow\quad \{$ definition of fhg , domains: (2) $\}$

$\quad ((\mathsf{fan}.H)G \circ \mathsf{fan}.G){>} \;\subseteq\; (\mathsf{fan}.H){>}$

$\Leftarrow\quad \{$ domains: (2) $\}$

$\quad ((\mathsf{fan}.H)G \circ \mathsf{fan}.G){>} \;\subseteq\; (H(\mathsf{fan}.G) \circ \mathsf{fan}.H){>}$

$\Leftarrow\quad \{$ monotonicity $\}$

$\quad (\mathsf{fan}.H)G \circ \mathsf{fan}.G \;\subseteq\; H(\mathsf{fan}.G) \circ \mathsf{fan}.H$

$=\quad \{$ $\mathsf{fan}.H$ has type $H \hookleftarrow \mathsf{Id}$ $\}$

$\quad \mathsf{true}$.

□

LEMMA 38.

$$F\perp\!\!\!\perp \circ fhg \circ (\mathsf{fan}.H)^{\cup} \circ \mathsf{fan}.H \;\;\subseteq\;\; F\perp\!\!\!\perp \circ gfh \;\;.$$

Proof

$\quad F\perp\!\!\!\perp \circ fhg \circ (\mathsf{fan}.H)^{\cup} \circ \mathsf{fan}.H$

$\subseteq\quad \{$ (8) with $R := fhg \circ (\mathsf{fan}.H)^{\cup}$ $\}$

$\quad F\perp\!\!\!\perp \circ (\mathsf{fan}.F)H \circ \mathsf{fan}.H$

$=\quad \{$ F distributes over composition,
$\perp\!\!\!\perp$ is zero of composition $\}$

$\quad F\perp\!\!\!\perp \circ F(\mathsf{fan}.G)H \circ (\mathsf{fan}.F)H \circ \mathsf{fan}.H$

$=\quad \{$ definition $\}$

$\quad F\perp\!\!\!\perp \circ gfh$.

□

THEOREM 39.

$$ghf \;\cap\; fhg \;\;=\;\; fgh \;\;.$$

Proof We have:

$\quad ghf \;\cap\; fhg \;\;=\;\; fgh$

$=\quad \{$ basic inclusions (see section 3.2) $\}$

$\quad ghf \;\cap\; fhg \;\subseteq\; fgh$

$=\quad \{$ $F\mathsf{id} = (\mathsf{mem}.F){>} \cup F\perp\!\!\!\perp$, distributivity $\}$

$\quad (\mathsf{mem}.F){>} \circ (ghf \cap fhg) \;\subseteq\; fgh$
$\quad \wedge \quad F\perp\!\!\!\perp \circ (ghf \cap fhg) \;\subseteq\; fgh$

$=\quad \{$ lemma 35 and lemma 36 $\}$

$\quad \mathsf{true}$.

□

6. Conclusion

In this paper, we have used relation algebra to formulate and reason about shape properties of datatypes. We have shown that shape selectors are closed under set intersection, and we have analysed in detail the properties of a collection of ”fans”. Particularly striking is how simple and effective the basis is for our calculations. There are just three components: relators, membership and (point-free) relation algebra. The notion of a relator is very simple because it involves just four simple algebraic properties (relators preserve identities and distribute through composition —i.e. are functors— and are monotonic and commute with converse). The notion of membership is yet simpler. The point-free relation algebra is, of course, very rich and so harder to master. Yet, as we have shown, calculations within the algebra are compact and easy to check. This is why we believe that it does provide a practical basis for the practising programmer. Other publications that consider the shape of datatypes (eg. [12, 13]) are based on representing relations as pullbacks in a category. We look forward to seeing how such formulations of shape properties are able to express the ideas in this paper. In a companion paper [4], the authors consider the shape-preservation properties of generic zips [10, 5]. Again, we look forward to seeing whether or not other formulations of shape properties are able to formulate and reason about zips as effectively and at the same level of generality.

References

[1] R.C. Backhouse, P. de Bruin, P. Hoogendijk, G. Malcolm, T.S. Voermans, and J. van der Woude. Polynomial relators. In M. Nivat, C.S. Rattray, T. Rus, and G. Scollo, editors, *Proceedings of the 2nd Conference on Algebraic Methodology and Software Technology, AMAST'91*, pages 303–326. Springer-Verlag, Workshops in Computing, 1992.

[2] R.C. Backhouse, H. Doornbos, and P. Hoogendijk. A class of commuting relators. Available via World-Wide Web at `http://www.cs.nott.ac.uk/~rcb/MPC/papers`, September 1992.

[3] R.C. Backhouse, T.S. Voermans, and J. van der Woude. A relational theory of datatypes. Available via World-Wide Web at `http://www.cs.nott.ac.uk/~rcb/MPC/papers`, December 1992.

[4] Roland Backhouse, Richard Bird, and Paul Hoogendijk. On shape-preservation properties of zips. In preparation.

[5] Roland Backhouse and Paul Hoogendijk. Generic properties of datatypes. In Roland Backhouse and Jeremy Gibbons, editors, *Generic Programming*, volume 2793 of *LNCS Tutorial Series*, pages 98–135. Springer, 2003.

[6] Richard S. Bird. Zippy tabulations of recursive functions. In Philippe Audebaud and Christine Paulin-Mohring, editors, *Mathematics of Program Construction, 9th International Conference, MPC2008, Marseille, France*, volume LNCS 5133, pages 92–109. Springer, 2008.

[7] Richard S. Bird and Oege de Moor. *Algebra of Programming*. Prentice-Hall International, 1997.

[8] P.J. Freyd and A. Ščedrov. *Categories, Allegories*. North-Holland, 1990.

[9] Paul Hoogendijk. *A Generic Theory of Datatypes*. PhD thesis, Department of Mathematics and Computing Science, Eindhoven University of Technology, 1997.

[10] Paul Hoogendijk and Roland Backhouse. When do datatypes commute? In Eugenio Moggi and Giuseppe Rosolini, editors, *Category Theory and Computer Science, 7th International Conference*, volume 1290 of *LNCS*, pages 242–260. Springer-Verlag, September 1997.

[11] Paul Hoogendijk and Oege de Moor. Container types categorically. *Journal of Functional Programming*, 10(2):191–225, 2000.

[12] C.B. Jay. A semantics for shape. *Science of Computer Programming*, 25(251–283), 1995.

[13] Neil Ghani Michael Abbott, Thorsten Altenkirch. Categories of containers. In *Foundations of Software Science and Computation Structures*, volume 2620 of *LNCS*, pages 23–38. Springer, 2003.

[14] J. Riguet. Relations binaires, fermetures, correspondances de Galois. *Bulletin de la Société Mathématique de France*, 76:114–155, 1948.