Article

Sheaf Mereology: Parts and Wholes in a Topos-Theoretic Setting

Firstname Lastname [1], Firstname Lastname [2] and Firstname Lastname [2,*]

[1] Affiliation 1; e-mail@e-mail.com
[2] Affiliation 2; e-mail@e-mail.com
* Correspondence: e-mail@e-mail.com; Tel.: (optional; include country code; if there are multiple corresponding authors, add author initials) +xx-xxxx-xxx-xxxx (F.L.)

Abstract

A single paragraph of about 200 words maximum. For research articles, abstracts should give a pertinent overview of the work. We strongly encourage authors to use the following style of structured abstracts, but without headings: (1) Background: place the question addressed in a broad context and highlight the purpose of the study; (2) Methods: describe briefly the main methods or treatments applied; (3) Results: summarize the article's main findings; (4) Conclusions: indicate the main conclusions or interpretations. The abstract should be an objective representation of the article, it must not contain results which are not presented and substantiated in the main text and should not exaggerate the main conclusions.

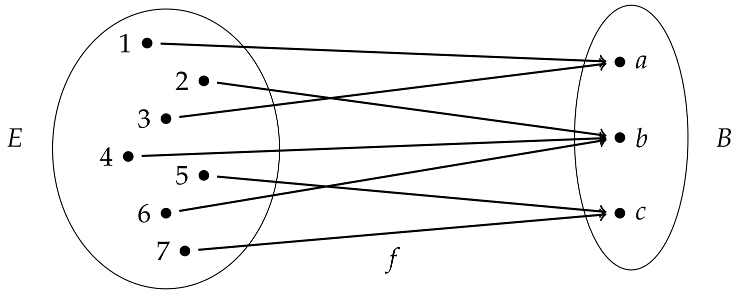
Keywords: mereology; fusions and integral wholes; sheaves; point-free topology; frames and locales; toposes; modality; merology logic; categorical logic

1. Sheaf-Theory

In this section, we cover the parts of sheaf theory that we will utilize in the rest of the paper. Readers familiar with sheaf theory can skip this section.

1.1. Fibers

Suppose we have a map (function) $f : E \to B$ that looks something like this:

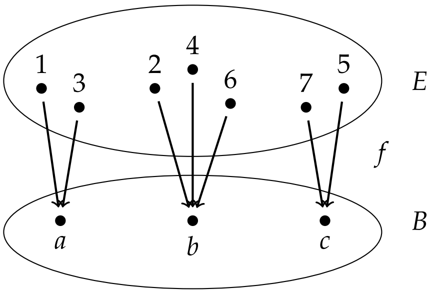


It is sometimes convenient to turn the diagram sideways and group together points in the domain that get sent to the same target, like so:

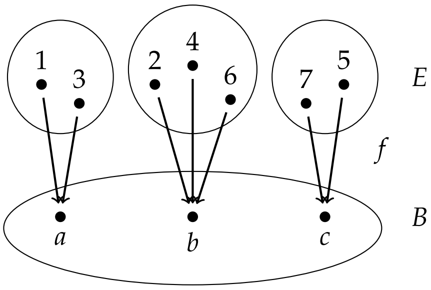
Received:
Revised:
Accepted:
Published:

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Philosophies* **2025**, *1*, 0. https://doi.org/

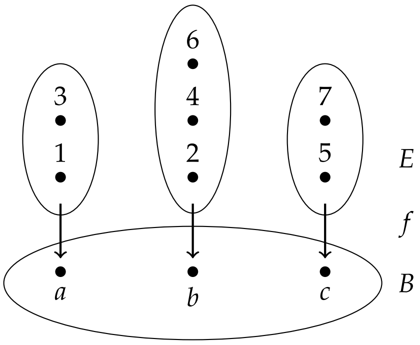
https://doi.org/10.3390/philosophies1010000



That makes the pre-images very easy to see. For any point in B, its pre-image is just the group of points sitting "over" it:



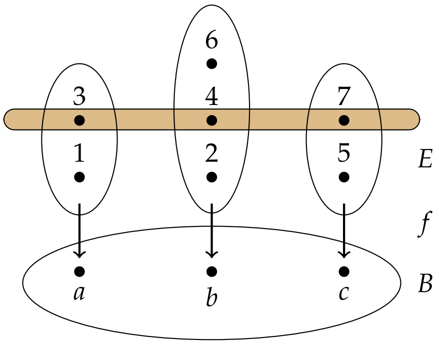
If we stack the points in each pre-image vertically, one on top of the other, we can then think of each pre-image as a kind of "stalk" growing over its base point:



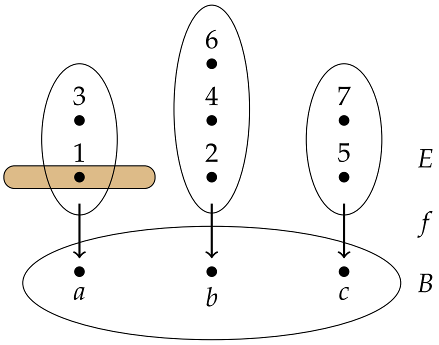
This gives rise to the idea of the "fibers" of a map. The fibers of a map are just its pre-images. For instance, the fiber over b is $\{2, 4, 6\}$.

Definition 1 (Fibers)**.** *Given a map $f : E \to B$ and a point $y \in B$, the fiber over y is its pre-image $f^{-1}(y) = \{x \mid f(x) = y\}$. B is called the "base space" (or "base" for short) of f, while the point y is called the "base point" (or "base" for short) of the fiber.*

We can select a cross-section of one or more fibers by selecting a point from each of the fibers in question. For instance, we can take 3, 4, and 7 as a cross-section of the fibers $f^{-1}(a)$, $f^{-1}(b)$, and $f^{-1}(c)$:



We can also take cross-sections local to only some of the fibers. For instance, we can take 1 as a cross-section just of $f^{-1}(a)$:



Definition 2 (Sections). *Given a map $f : E \to B$ and a subset $C \subseteq B$ (i.e., a selection of base points in B), a section of f (over C) is a choice of one element from each fiber over each base $x \in C$.*

Remark 1. *Since each point in a fiber amounts to a section over the fiber's base, the elements of a fiber are often just called the "sections" of the fiber.*

1.2. Spaces

In the above examples, the base B was a set. We often want to consider bases that have more structure, e.g., bases that have spatial structure.

In traditional topology, spaces are built out of the points of the space. Given a set of points, a topology on that set specifies which points belong in which regions of the space.

Definition 3 (Topology). *Let X be a non-empty set, thought of as the "points" of the space. A topology on X is a collection T of subsets of X, thought of as the "regions" of the space (called the "open sets" or just the "opens" of T), that satisfy the following conditions:*

(T1) The empty set and the whole set are open:

$$\varnothing \in T, X \in T.$$

(T2) Arbitrary unions of opens are open:

$$\textit{if } \{U_i\}_{i \in I} \subseteq T, \textit{ then } \bigcup_{i \in I} U_i \in T.$$

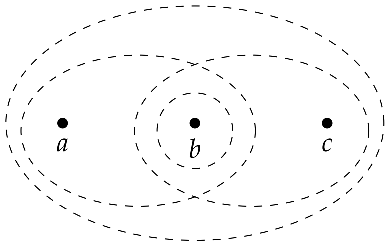
(T3) Finite intersections of opens are open:

$$\textit{if } U_1, \ldots, U_n \in T, \textit{ then } \bigcap_{i=1}^{n} U_i \in T.$$

These conditions encode the way that spatial regions are put together. For instance, it ensures that if two regions overlap, then their overlapping area is a region too (and indeed, that's what it means for regions to *overlap*: there's a region of space they have in common).

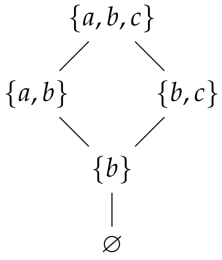
Remark 2. *The regions of a topology form a (join-)complete lattice. T1 says that the lattice has a top ($\top$) and bottom ($\bot$), T2 says that for any selection of opens there is a join, and T3 says that for any finite selection of opens there is a meet.*

Example 1. *Let $X = \{a, b, c\}$. One possible topology is: $T = \{\varnothing, \{b\}, \{a, b\}, \{b, c\}, \{a, b, c\}\}$. If we draw dashed circles around each of the opens (regions), ignoring the empty set, we get:*



There are two regions $\{a, b\}$ and $\{b, c\}$ that overlap at b (so $\{b\}$ is a region in T too). There is also the full region $\{a, b, c\}$, which is the union of the smaller regions.

We can draw T as a Hasse diagram, which shows that the regions form a lattice:



The lattice structure suggests that much of what is important about a space is not so much its points, but rather its opens/regions. This leads to the idea that topology-like reasoning can be done without the points. So, we can generalize: take a topology, and drop the points. That leaves just the opens/regions, which we call a frame (or locale).

Definition 4 (Frames/locales). *A frame (synonymously, a locale) $\mathbb{L}$ is a partially ordered set L (we call its elements "opens" or "regions") that satisfies the following conditions:*

(L1) L is a complete lattice:

- *Every subset $S \subseteq L$ has a join, denoted $\bigvee S$.*
- *Every finite subset $F \subseteq L$ has a meet, denoted $\bigwedge F$.*

(L2) Finite meets distribute over arbitrary joins:

$$a \wedge \bigwedge_{i \in I} b_i = \bigwedge_{i \in I} (a \wedge b_i), \text{ for all } a \in L \text{ and all families } \{b_i\}_{i \in I} \subseteq L.$$

Define $V \preccurlyeq U$ (read "V is included in U") by $a = a \wedge b$.

Remark 3. *The fact that $V \preccurlyeq U$ is equivalent to $a = a \wedge b$ means we can deal with the opens of a frame algebraically (via $\wedge$ and $\vee$ operations), or order-theoretically (via the $\preccurlyeq$ relation), whichever is more convenient.*

Remark 4. *The category of locales is defined as the dual/opposite of the category of frames (see* **??** *below), and so frames and locales are quite literally the very same objects. In practice, frames are often used for algebraic purposes, and locales are used for (generalized) spatial purposes. Here, we will have no reason to distinguish these two roles, and so we will use the names "frame" and "locale" interchangeably.*

1.3. Presentations of locales

Locales have presentations much like groups and other algebraic structures. To give the presentation of a locale, specify a set of generators and relations.

Definition 5 (Presentations). *A presentation $\langle G, R \rangle$ of a locale $\mathbb{L}$ is comprised of:*

(P1) A set of generators $G = \{U_k, U_m, \ldots\}$.
(P2) A set of relations $R \subseteq G \times G$ *on those generators.*

The locale $\mathbb{L}$ *presented by* $\langle G, R\rangle$ *is the smallest one freely generated from* G *which satisfies* R.

To calculate the locale that corresponds to a presentation, start with the generators, then take all finite meets and all arbitrary joins that satisfy R (and of course L1 and L2).

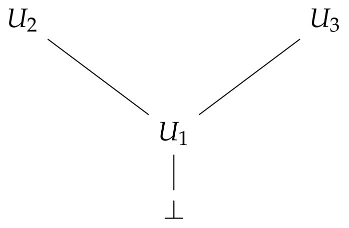
Example 2. *Let a locale* $\mathbb{L}$ *be given by the presentation* $\langle G, R\rangle$ *where:*

- $G = \{\perp, U_1, U_2, U_3\}$.
- $R = \{\perp \preccurlyeq U_1, U_1 \preccurlyeq U_2, U_1 \preccurlyeq U_3\}$.

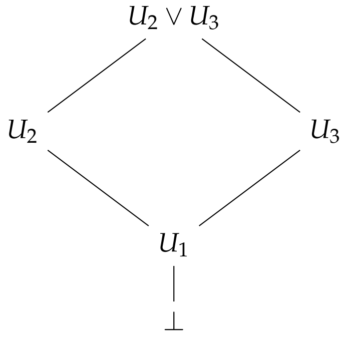
There are four generators ($\perp$*,* U_1*,* U_2*, and* U_3*), and* $\perp$ *is below* U_1 *while* U_1 *is a sub-region of* U_2 *and* U_3*. Since* U_1 *is a sub-region of both* U_2 *and* U_3*,* U_1 *is their meet:*

- $U_1 = U_2 \wedge U_3$.

At this point, we have generated this much of the locale:



R *says nothing to constrain joins, so we need to join everything we can. In this case, we need to join* U_2 *and* U_3*:*



There are no further joins or meets that aren't already represented in the picture. For instance, all further non-trivial meets are already accounted for:

- $U_1 \wedge \perp = \perp$.
- $U_2 \wedge U_1 = U_1$ *and* $U_3 \wedge U_1 = U_1$.
- $U_2 \wedge \perp = \perp$ *and* $U_3 \wedge \perp = \perp$.
- $(U_2 \vee U_3) \wedge U_2 = U_2$ *and* $(U_2 \vee U_3) \wedge U_3 = U_3$.
- $(U_2 \vee U_3) \wedge U_1 = U_1$.
- $(U_2 \vee U_3) \wedge \perp = \perp$.

Similarly, all other non-trivial joins are also already accounted for:

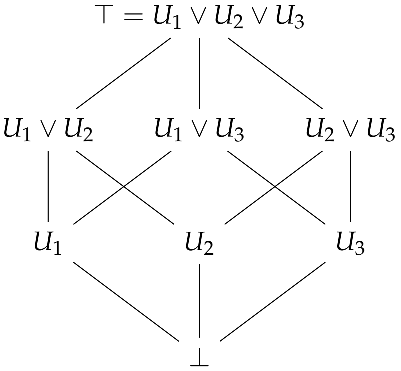
- $\perp \vee U_1 = U_1$.
- $\perp \vee U_2 = U_2$ *and* $\perp \vee U_3 = U_3$.
- $\perp \vee (U_2 \vee U_3) = U_2 \vee U_3$.
- $U_1 \vee U_2 = U_2$ *and* $U_1 \vee U_3 = U_3$.

- $U_2 \vee (U_2 \vee U_3) = U_2 \vee U_3$ *and* $U_2 \vee (U_3 \vee U_3) = U_2 \vee U_3$.

Example 3. *Let* $\mathbb{L} = \langle G, R \rangle$ *be given by:*

- $G = \{U_1, U_2, U_3\}$.
- $R = \varnothing$.

We have three generators (U_1, U_2, and U_3), and there are no relations restricting how those generators are related. Thus, the locale that is freely generated from this presentation is isomorphic to the power set of three elements:



A presentation provides the most "minimal" information from which the rest of the locale is generated. It might be tempting to think that each generator is an atomic region, but that is not quite right. Some generators are reducible to others.

Definition 6 (Meet-irreducibility)**.** *Given a presentation* $\mathbb{L} = \langle G, R \rangle$*, a region* $U \in G$ *is meet-reducible ("reducible" for short) if it is the non-trivial meet of other regions:*

$$U = W \wedge V \implies (V = U \textit{ or } W = U).$$

U is meet-irreducible ("irreducible" for short) if it is not meet-reducible.

Intuitively, a generator is reducible if it can be expressed as the meet of strictly larger regions, which occurs exactly when it is their overlap.

Example 4. *Take the locale from Example 2.* U_1 *is the overlap of* U_2 *and* U_3*, and* U_2 *and* U_3 *are strictly larger regions than* U_1*, so* U_1 *is reducible.*

By contrast, U_2 *and* U_3 *are irreducible, because they cannot be expressed as the meet of two strictly larger regions. Similarly,* $\bot$ *is irreducible, because it is not the meet of two strictly larger regions either (it is the meet of only one strictly larger region, namely* U_1*).*

We can see the minimal irreducible generators of a locale as its atomic regions.

Definition 7 (Atomic regions)**.** *Given a presentation* $\mathbb{L} = \langle G, R \rangle$*, define the atomic regions of* $\mathbb{L}$*, denoted* $Atoms(\mathbb{L})$*, as the minimal irreducible generators, i.e. those generators* $g \in G$ *that satisfy the following two conditions:*

(A1) Meet-irreducibility*.* g *is meet-irreducible.*
(A2) Minimality*. There is no strictly smaller meet-irreducible* h *with* $h \preccurlyeq g$*.*

1.4. Presheaves

Above we considered the fibers of a map $f : E \to B$, where E and B were sets. We can also consider fibers over locales, where the fibers respect the locale's structure. This is called a presheaf. A presheaf is an assignment of data to each of a locale's regions that is "stable under restriction," i.e., that respects zooming in and out.

Definition 8 (Presheaf)**.** *Let $\mathbb{L}$ be a locale, and let $Morphs(\mathbb{L})$ be $\{\langle A, B\rangle \mid A \preccurlyeq B \in \mathbb{L}\}$. A presheaf on $\mathbb{L}$ is a pair $\langle F, \{\rho^B_A\}_{\langle A,B\rangle \in Morphs(\mathbb{L})}\rangle$, where:*

- *F assigns to each region $U \in L$ some data $F(U)$.*
- *$\{\rho^B_A\}_{\langle A,B\rangle \in Morphs(\mathbb{L})}$ is a family of maps $\rho^B_A : F(B) \to F(A)$ (called "restriction maps"), each of which specifies how to restrict the data over $F(B)$ down to the data over $F(A)$.*

All together, $\langle F, \{\rho^B_A\}_{\langle A,B\rangle \in Morphs(\mathbb{L})}\rangle$ must satisfy the following conditions:

(R1) Restrictions preserve identity:

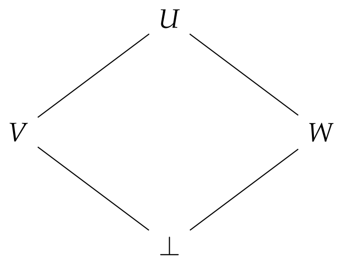
$$\rho^U_U = id_U \text{ (the identity on } U\text{), for every } U \in \mathbb{L}.$$

(R2) Restrictions compose:

$$\text{If } A \preccurlyeq B \text{ and } B \preccurlyeq C, \text{ then } \rho^C_A = \rho^B_A \circ \rho^C_B.$$

Since F assigns data $F(U)$ to each region $U \in \mathbb{L}$, we can think of the $F(U)$s as the "fibers" over $\mathbb{L}$, and the restriction maps as "zoom in" maps that go from bigger fibers down to smaller fibers.

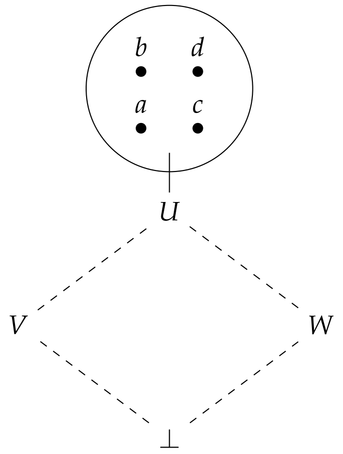
Example 5. *Let $\mathbb{L}$ be a locale $\{\bot, W, V, U\}$ with the following structure:*



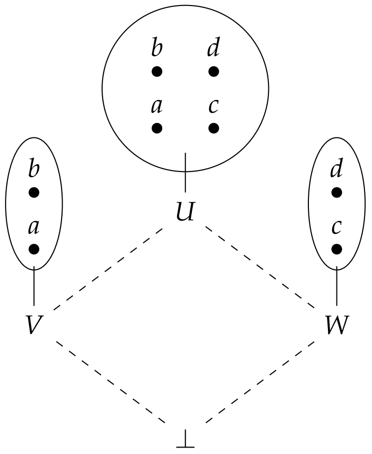
Next, let's define a presheaf F as follows:

- *$F(U) = \{a, b, c, d\}$, $F(V) = \{a, b\}$, $F(W) = \{c, d\}$, $F(\bot) = \{*\}$.*
- *Define ρ^U_V as the projection (send a to a, b to b, and the rest can go anywhere), and similarly for ρ^U_W. Let $\rho^U_\bot$, $\rho^V_\bot$, and $\rho^W_\bot$ send their data to $\{*\}$, and let the rest be identities.*

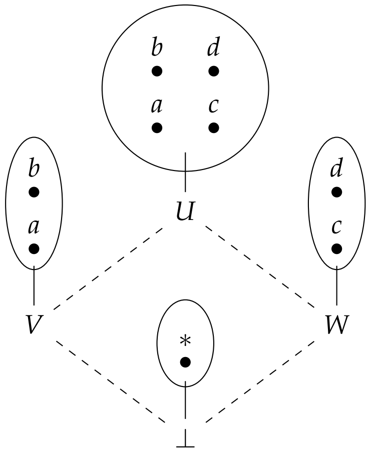
We can see F's assignments as fibers over $\mathbb{L}$ by drawing them over the regions they are assigned to. For instance, over U we have $F(U)$, i.e., $\{a, b, c, d\}$:



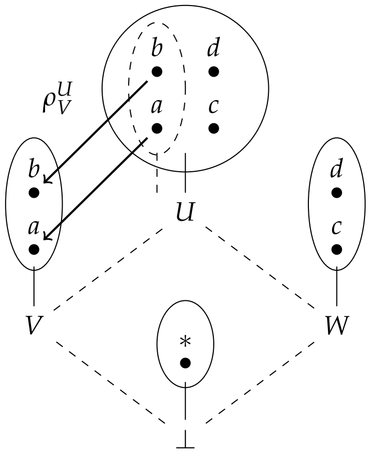
Similarly, over V and W, we have $F(V) = \{a, b\}$ and $F(W) = \{c, d\}$:



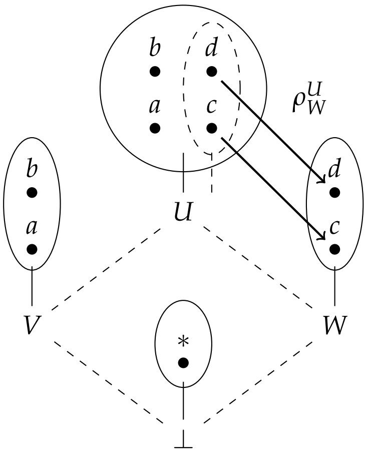
Finally, over $\perp$, we have a singleton set:



The restriction maps show how to "zoom in" on the parts (sub-fibers) of any given fiber. For instance, we can see that the fiber over V is contained in the fiber over U. The restriction map just projects that sub-fiber out, thereby showing us how to "zoom in" on that sub-fiber:

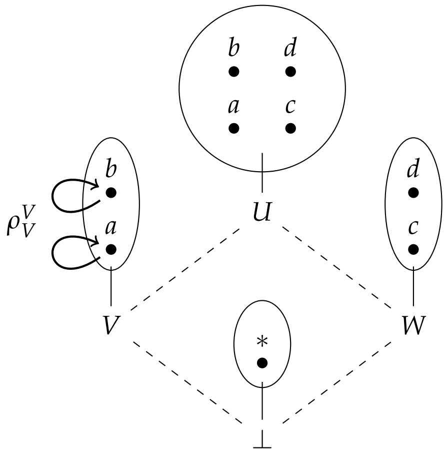


It's similar for the fiber over W:



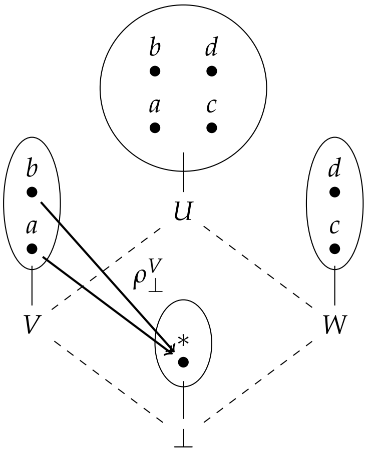
178

Restricting a fiber to itself is just the identity on the fiber: 179



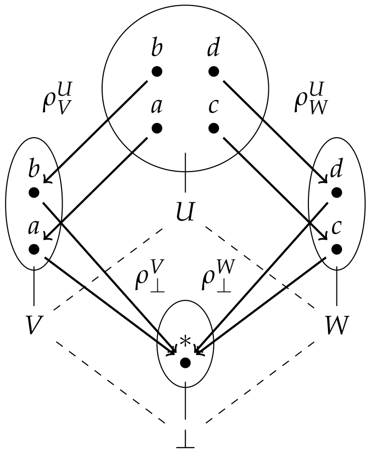
180

The other restriction maps restrict down to the singleton set. For instance: 181



182

All of this makes it clear that the structure of the presheaf data that sits in the fibers over $\mathbb{L}$ mimics (respects) the structure of the base locale: 183 184



Remark 5. *In line with Remark 1, the elements of each fiber $F(U)$ are usually just called the "sections" of $F(U)$. For instance, c is a section of $F(W)$, as is d.*

1.5. Sheaves

The definition of a presheaf requires only that the data be stable under restriction (zooming in on a region). It does not require that the data fit together across different regions (fibers).

A sheaf is a presheaf with an added gluing condition: whenever you have compatible data on overlapping fibers, there must be a unique way to glue it together into data over the union. In other words, the data in the fibers must agree on overlap and combine coherently.

To get at this idea, let's first define a cover. A cover of a region U is a selection of sub-regions that covers U in its entirety. The chosen sub-regions don't leave any part of U exposed.

Definition 9 (Cover)**.** *Let $\mathbb{L}$ be a topology or a locale, and let U be a region of $\mathbb{L}$. A cover of U is a family $\{U_i\}_{i\in I} \subseteq \mathbb{L}$ such that:*

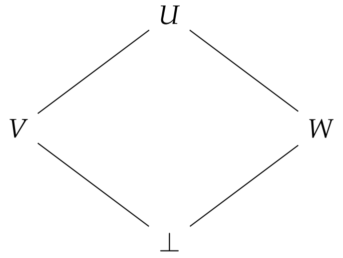
$$U = \bigvee_{i\in I}\{U_i\}.$$

In other words, a cover of U is a family of regions that join together to form U.

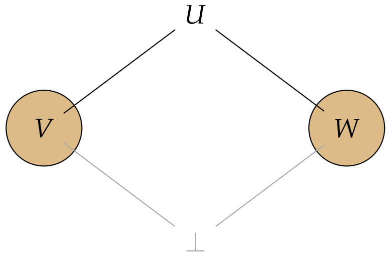
Example 6. *Take the topology from Example 1: $T = \{\varnothing, \{b\}, \{a,b\}, \{b,c\}, \{a,b,c\}\}$. A cover of $\{a,b,c\}$ is $\{a,b\}$ and $\{b,c\}$, because altogether, $\{a,b\}$ and $\{b,c\}$ cover all of the points in $\{a,b,c\}$.*

Another cover of $\{a,b,c\}$ is $\{\{a,b\},\{b,c\},\{b\}\}$. Although $\{b\}$ is redundant here, this choice of sub-regions still entirely covers $\{a,b,c\}$ as required.

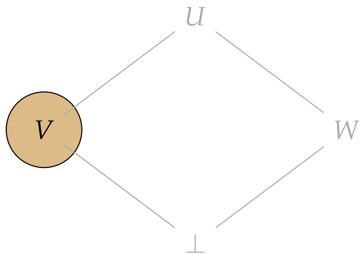
Example 7. *In the context of frames, where there are no points, a cover of U is just a selection of sub-regions of U that together join together to form U. Take the locale from Example 5:*



A cover of U is $\{V,W\}$, since $U = \bigvee\{V,W\}$:



A cover of V is just $\{V\}$:



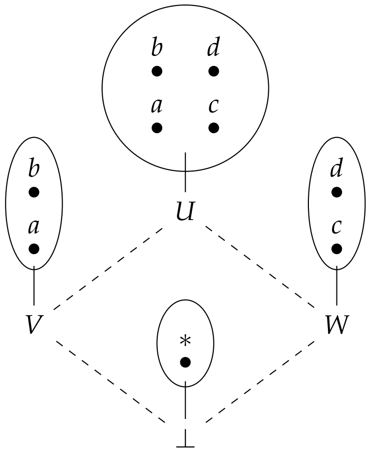
Remark 6. *A cover over the least element of a locale (or a topology) is empty (the empty set), because there are no regions (or points) to cover.*

Given a presheaf F over a locale $\mathbb{L}$, if we have a cover $\{U_i\}_{i\in I}$ of some portion of $\mathbb{L}$, there is a corresponding family of fibers $\{F(U_i)\}_{i\in I}$ over that cover. We can pick one section (i.e., one element) from each such fiber to get a slice of elements that spans all of the fibers over that cover. Let us call such a choice a selection of patch candidates.

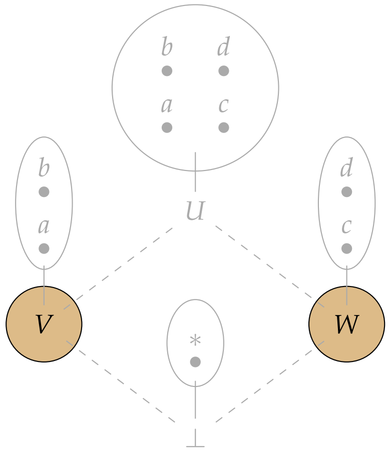
Definition 10 (Patch candidates)**.** *Given a presheaf F and a cover $\{U_i\}_{i\in I}$ with a corresponding family of fibers $\{F(U_i)\}_{i\in I}$, a selection of patch candidates $\{s_i\}_{i\in I}$ is a choice of one section s_i from each $F(U_i)$:*

$$\{s_i\}_{i\in I} = \{s_i \mid s_i \in F(U_i) \text{ for each } F(U_i) \in \{F(U_i)\}_{i\in I}\}.$$

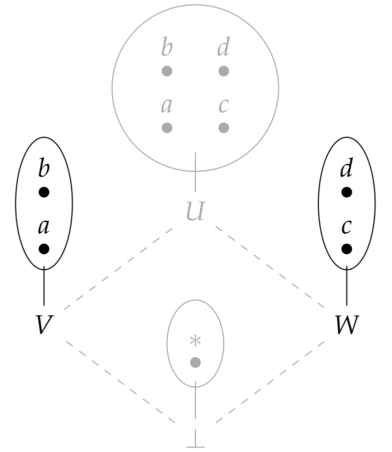
Example 8. *Take the presheaf from Example 5:*



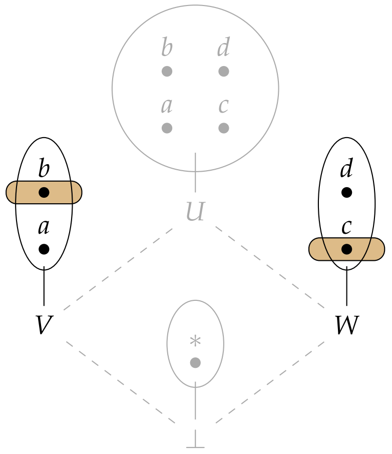
Let $\{V, W\}$ be the cover of interest:



Over this cover, we have a corresponding family of fibers:



A selection of patch candidates is a choice of one section (element) from each fiber. For instance, we might pick b from $F(V)$ and c from $F(W)$:



Similarly, we might pick $\{a, d\}$, $\{b, d\}$, or $\{a, c\}$, each of which is a valid selection of patching candidates.

Remark 7. *Consider the empty cover. Since there are no sub-regions below the least element of a locale, there are no patch candidates we could choose for the empty cover either. Hence, any selection of patch candidates for the empty cover is $\varnothing$.*

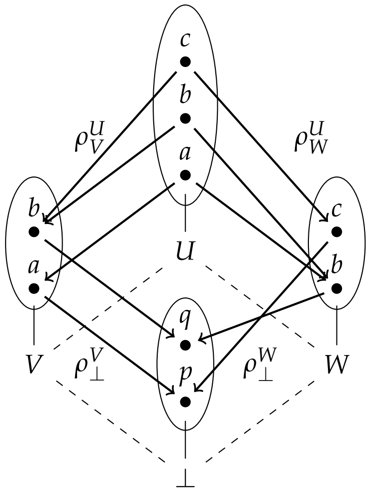
A selection of patch candidates might fit together, or they might not. We say they are compatible if they fit together, i.e., if they agree on overlaps. To check this, take any pair of patch candidates, and check if they restrict to the same data on their overlap.

Definition 11 (Compatible patch candidates)**.** *Given two fibers $F(U_i)$ and $F(U_j)$ and a patch candidate from each, $s_i \in F(U_i)$ and $s_j \in F(U_j)$, s_i and s_j are compatible if they restrict to the same data on their overlap $U_i \wedge U_j$:*

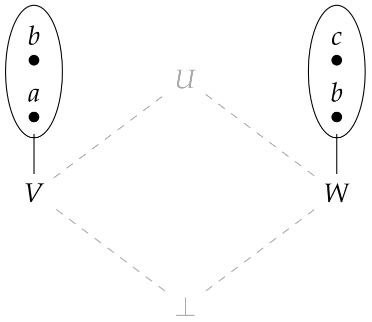
$$\rho^{U_i}_{U_i \wedge U_j}(s_i) = \rho^{U_j}_{U_i \wedge U_j}(s_j).$$

A selection of patch candidates $\{s_i\}_{i \in I}$ is compatible if all of its members are pair-wise compatible.

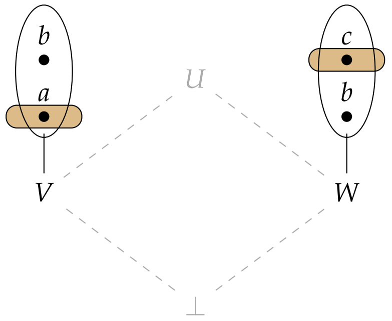
Example 9. *Consider the following presheaf F:*



Take the cover $\{V, W\}$ and its corresponding fibers:

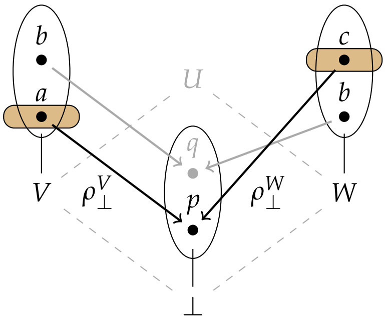


Suppose we pick $\{a, c\}$ for patch candidates:

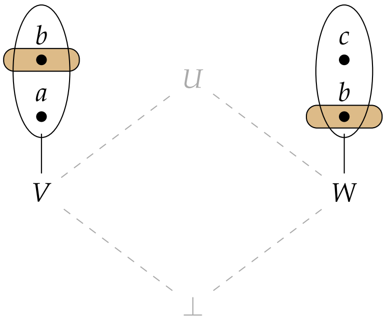


Is this selection compatible? We have to check if they agree on their overlap. The overlap $V \wedge W$ is $\perp$. Where does $\rho^V_\perp$ send our chosen patch candidate a? It sends it to p, since $\rho^V_\perp(a) = p$.

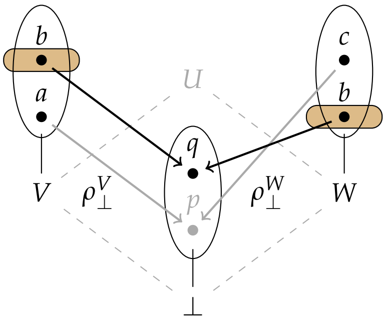
Where does $\rho_\perp^W$ send our other chosen patch candidate b? It also sends it to p, since $\rho_\perp^W(c) = p$. On the overlap $\perp$ then, $\rho_\perp^V(a) = \rho_\perp^W(c)$, so a and c are compatible. This is easy to see in the diagram, since a and b both get sent to the same place:



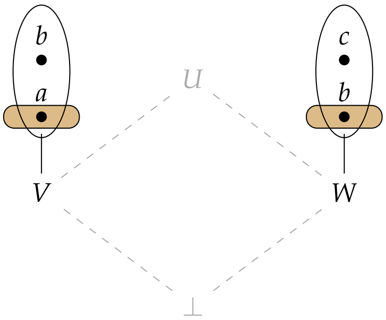
Now suppose we pick $\{b, b\}$ for patch candidates:



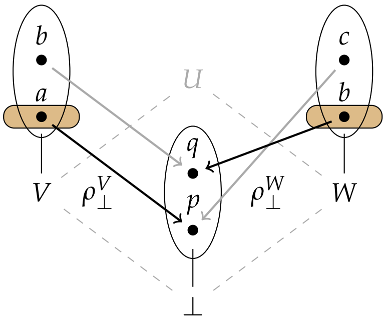
These are also compatible. They agree on their overlap (both restrict to q):



Finally, suppose we pick $\{a, b\}$ for patch candidates:



These are not compatible. They do not agree on their overlap:



Remark 8. *Consider the empty cover. Since any selection of patch candidates for the empty cover is empty, compatibility is satisfied vacuously.*

As an analogy, if you ask your class to turn off all cell phones but nobody brought a cell phone to glass, then your request is satisfied vacuously: there is simply nothing that needs to be done to make it happen. It's similar with the empty cover: since there are patch candidates to check, compatibility is achieved vacuously.

If a selection of patch candidates $s_i, \ldots, s_k$ across a cover of U is compatible, we say those patches glue together if if there's a section s in $F(U)$ that restricts down to exactly those patches.

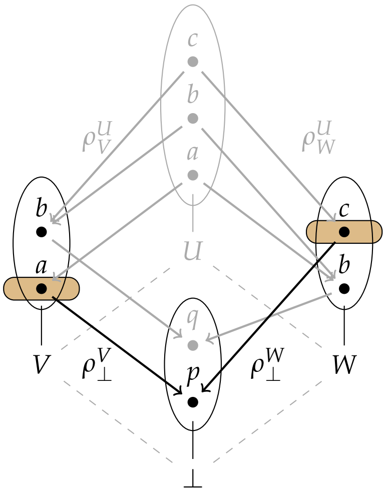
Definition 12 (Gluing). *Given a presheaf F and a selection of compatible patch candidates* $\{s_i\}_{i\in I}$ *for a cover* $\{U_i\}_{i\in I}$, $\{s_i\}_{i\in I}$ *glue together only if there is a section* $s \in F(U)$ *that restricts down to* s_i *on each fiber* $F(U_i)$ *of the cover, i.e., only if s is such that:*

$$\rho^U_{U_i}(s) = s_i, \text{ for each } i \in I.$$

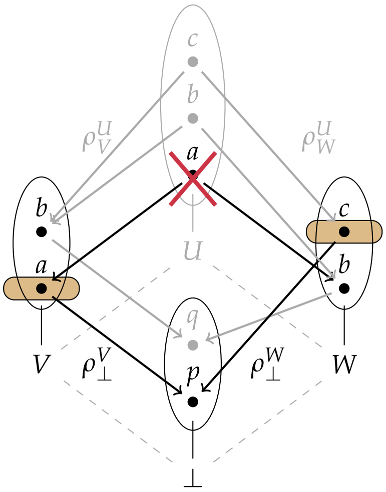
As a matter of terminology, if a section $s \in F(U)$ *is glued from patches* $\{s_i\}_{i\in I}$*, we say that s is a global section of the cover, and each* s_i *is a local section of the cover. We may also say variously that s is* glued *from those patches, that s is* composed *of those pathes, that those patches* compose *s, or that gluing those patches* yields *s.*

A selection of patches $\{s_i\}_{i\in I}$ *glues uniquely if there is one and only one such section* $s \in F(U)$ *that is glued from them.*

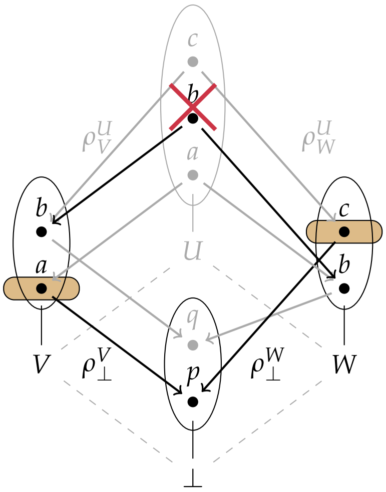
Example 10. *Take the presheaf from Example 9, and consider the cover* $\{V, W\}$ *again. Take the the selection of patches* $\{a, c\}$*, which are compatible because they agree on overlap:*



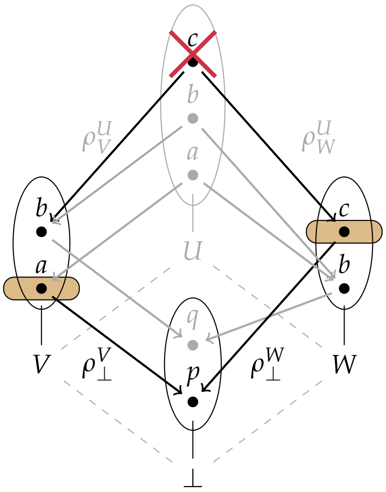
Even though a and c are compatible, they do not glue together, because there is no section in $F(U)$ *that restricts down to them. Consider* $a \in F(U)$ *first. It restricts to* $a \in F(V)$ *on the left, but it does not restrict to* $c \in F(W)$ *on the right:*



As for $b \in F(U)$*, it restricts to neither* $a \in F(V)$ *on the left nor* $c \in F(W)$ *on the right:*

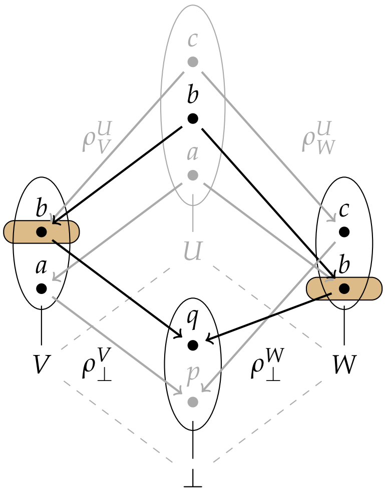


Finally, $c \in F(U)$ *restricts to* $c \in F(W)$ *on the right, but not to* $a \in F(V)$ *on the left:*



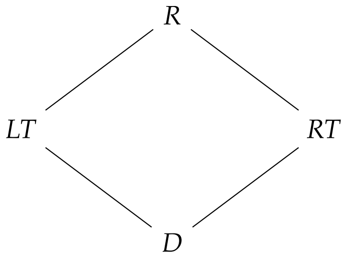
Thus, none of a, b, or c in $F(U)$ *are glued from* $\{a, c\}$*, because none of them decompose into a on the left and c on the right.*

Now suppose we pick $\{b, b\}$ for patch candidates. These do glue together (trivially), because there is a section in $F(U)$ (namely $b \in F(U)$) that restricts down to $b \in F(V)$ on the left and $b \in F(W)$ on the right:



Example 11. *Consider an example that glues together behaviors. Imagine a toy robot that looks something like a small tank: it has tracks on the left and right sides, and the two tracks are connected by a single drive controller. The controller either drives at a constant speed, or it sits idle. When it drives, it turns both tracks at the same speed.*

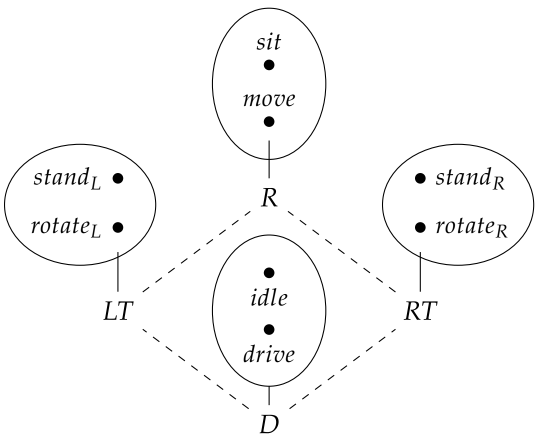
Let's represent the robot as a locale. Let LT and RT be the left and right track assemblies respectively, let D be the drive controller that is shared by LT and RT, and let R be the whole robot (the join of LT and RT). As a picture:



For a presheaf, let's assign to each region the behaviors that are locally observable at that region:

- *The drive controller D can either drive (turn) or sit idle.*
- *The left track assembly can each either rotate$_L$ or stand$_L$ still.*
- *The right track assembly can also either rotate$_R$ or stand$_R$ still.*
- *The entire robot can either move (forward) or sit stationary.*

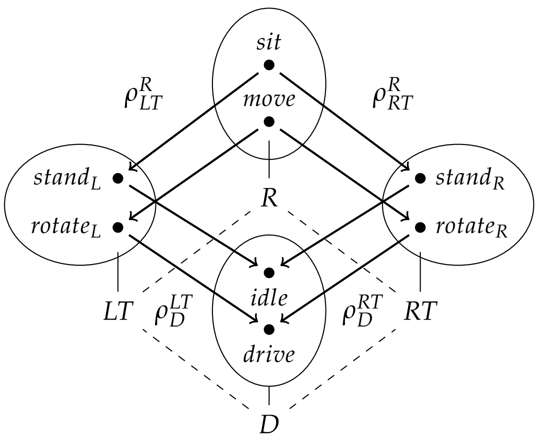
In a picture:



For the restriction maps, let's say that they restrict the observable behavior of a larger region to the observable behavior of the smaller region. For instance, if you are observing the whole robot moving forward (move), and you then "zoom in" on the left track assembly, you'll see those tracks rotating ($rotate_L$).

- $\rho^R_{LT}(sit) = stand_L$, $\rho^R_{LT}(move) = rotate_L$.
- $\rho^R_{RT}(sit) = stand_R$, $\rho^R_{RT}(move) = rotate_R$.
- $\rho^{LT}_D(stand_L) = idle$, $\rho^{LT}_D(rotate_L) = drive$.
- $\rho^{RT}_D(stand_R) = idle$, $\rho^{RT}_D(rotate_R) = drive$.

In a picture:



Now take the cover $\{LT, RT\}$ of R. The patch candidates $\{rotate_L, rotate_R\}$ are compatible, because they agree on overlap (they both restrict down to drive). But they also glue uniquely, yielding move. In other words, the robot's forward motion is patched together precisely from the two pieces of its cover, namely the left tracks rotating ($rotate_L$) and the right tracks rotating ($rotate_R$).

Similarly, the Robot's sit behavior is also glued from the two pieces of its cover, namely the left track assembly standing still ($stand_L$) and the right track assembly standing still ($stand_R$).

Thus, there are two global sections of R's behavior: moving forwards (patched together from its left and right motions), or standing still (patched together from its left and right lack of motion).

We can now state what it is to be a sheaf. A sheaf is a presheaf that satisfies a special gluing condition: namely, that for every cover, every compatible selection of patch candidates glues together uniquely.

Definition 13 (Sheaf)**.** *A presheaf F is a sheaf iff it satisfies the following condition (called "the gluing condition"):*

(G0) *For every cover* $\{U_i\}_{i\in I}$ *of a region* U *and every selection of patch candidates* $\{s_i\}_{i\in I}$ *for that cover, if* $\{s_i\}_{i\in I}$ *can glue, then* $\{s_i\}_{i\in I}$ *glue to yield a unique* $s \in F(U)$.

Remark 9. *There is a subtlety regarding what sheaves look like over the least element of a locale. Note that the gluing condition is formulated as an implication. That is to say, it says that, for every cross-section of patch candidates,* if *that cross-section can glue,* then *it glues in exactly one way.*

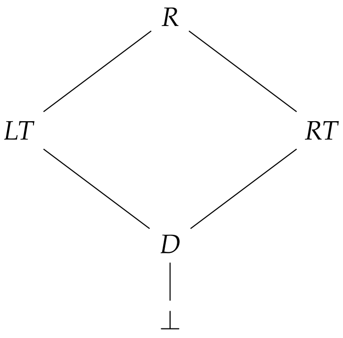
Next, consider the fact that the cover over the least region of a locale is an empty cover. Since there are no patch candidates that need to be checked for compatibility, there is nothing that needs to be done to get a "selection of gluable patch candidates." Hence, the antecedent of the gluing condition is satisfied vacuously over the least element of the locale.

But since the empty cover satisfies the antecedent of the gluing condition vacuously, it follows that if a presheaf is to qualify as a sheaf, it must ensure that the consequent is satisfied over the empty cover as well. In other words, it must assign a unique glued section (a singleton set) to the least region of the locale. So, even though a presheaf *may assign a larger set of data to the least element of a locale, a* sheaf *always assigns a singleton to that region.*

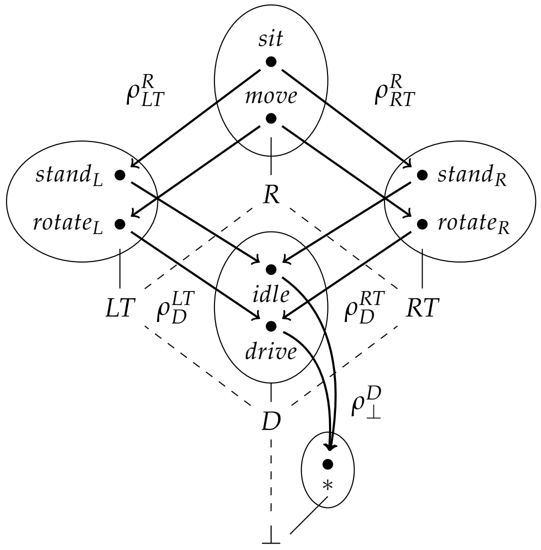
Example 12. *The presheaf from Example 9 fails to be sheaf, because as we saw in Example 10, there is a compatible selection of patch candidates (namely,* $\{a, c\}$*) which fails to glue. To be a sheaf, every compatible selection of patch candidates must glue.*

Example 13. *The presheaf from Example 11 fails to be a sheaf, because it does not assign a singleton to the lowest region of the underlying locale. In that example* D *is the lowest region of the locale, and* $F(D) = \{p, q\}$*, a set containing two elements. Hence,* F *fails to be a sheaf.*

However, suppose we add a distinct bottom element to the locale:



If we assign a singleton set to $\bot$ *(so that* $F(\bot) = \{*\}$*, say), then* F *looks like this:*



This modification ensures that F qualifies as a sheaf, since it ensures that all *gluable selections of patch candidates (including the empty one) glue uniquely.*

1.6. A Canonical Sheaf Construction

Not every presheaf is a sheaf, since some presheaves fail the gluing condition. However, there is a canonical procedure called "sheafification" that turns any presheaf into a sheaf. To sheafify a presheaf, add any missing sections that glue, then quotient sections that are locally indistinguishable. The result is guaranteed to be a sheaf, by construction.

For our purposes, there is a simplified version of sheafification that we can use to construct sheaves that model part-whole complexes in a natural way. Given a presentation of a locale, the recipe to build such a sheaf over it goes like this:

1. Assign local data to atomic regions.
2. Specify a gluing condition.
3. Recursively glue more and more pieces together until you can't glue any more.

Let's make this more precise. Given a presented locale, we can uniquely write each region as the join of its atomic regions.

Definition 14 (Atomic indices)**.** *Let $\mathbb{L} = \langle G, R \rangle$ be a presented locale with $G = \{U_1, \dots, U_n\}$. Let $\mathbb{A} \subseteq \{0, \dots, n\}$ be the indices of the atomic regions of G.*

For any $U \in \mathbb{L}$, define its atomic support (denoted $I(U)$, or just I for short) as:

$$I(U) = \{i \in \mathbb{A} \mid U_i \preccurlyeq U\}$$

Then U can be written canonically as $U_{I(U)}$, the join of its atomic supports:

$$U_{I(U)} = \bigvee_{i \in I(U)} U_i.$$

A gluing condition is a family of predicates that say when a selection of patch candidates glue.

Definition 15 (Gluing condition)**.** *A gluing condition $\mathcal{G}$ is a family of predicates*

$$\mathcal{G}_U : \prod_{i \in I(U)} F(U_i) \to \{\text{true}, \text{false}\},$$

one for each region $U \in \mathbb{L}$, that collectively satisfy the following coherence conditions:

(G1) Local data is glued. *If $U_k \in Atoms(\mathbb{L})$ and $F(U_k) = \{\langle b_k \rangle\}$, then*

$$\mathcal{G}_{U_k}(\langle b_k \rangle) = \text{true}.$$

(G2) Downward stability. *If $\mathcal{G}_U(\langle b_i \rangle_{i \in I(U)}) = \text{true}$, then for each $V \preccurlyeq U$, we must have:*

$$\mathcal{G}_V(\rho^U_V(\langle b_i \rangle_{i \in I(U)})) = \text{true}.$$

(G3) Upward stability. *Given a selection of patch candidates $\langle b_i \rangle_{i \in I(U)}$, if $\mathcal{G}_{U_i \vee U_j}(\langle b_i, b_j \rangle) =$ true for each $i, j \in I(U)$, then we must have:*

$$\mathcal{G}_U(\langle b_i \rangle_{i \in I(U)}) = \text{true}.$$

A sheaf can be generated from a gluing condition by starting with some local data on the atomic regions and then gluing all pieces together that satisfy the gluing condition.

Definition 16 ($\mathcal{G}$-sheaves)**.** *Given a gluing condition $\mathcal{G}$ and local data $F(U_k) = \{\langle b_k \rangle\}$ for each atomic region U_k, define for each region U:*

$$F(U) = \{\langle b_i \rangle_{i \in I(U)} \in \prod_{i \in I(U)} F(U_i) \mid \mathcal{G}_U(\langle b_i \rangle_{i \in I(U)}) = \text{true}\}.$$

For $V \preccurlyeq U$, define the restriction map

$$\rho^U_V : F(U) \to F(V) \quad \textit{as} \quad \rho^U_V(\langle b_i \rangle_{i \in I(U)}) = \langle b_i \rangle_{i \in I(V)}.$$

Set $F(\bot) = \{\langle\rangle\}$, the empty tuple.

Remark 10. *Alternatively, given some local data and a gluing condition, define a presheaf over the given locale, call it $F_\wp$, that assigns all combinations of local data to each region:*

$$F_\wp(U) = \prod_{i \in I(U)} F(U_i).$$

Then filter by the gluing condition. That produces the same sheaf.

We must check that Definition 16 really defines a sheaf.

Theorem 1 ($\mathcal{G}$-sheaves are presheaves)**.** *Given a gluing condition $\mathcal{G}$ and an assignment of local data to the atomic regions of the underlying locale, the corresponding $\mathcal{G}$-sheaf is a presheaf.*

Proof. We must show that restrictions preserve identities and composition.

- *Identities.* ρ^U_U projects to the same index set, so $\rho^U_U = id_{F(U)}$.
- *Composition.* ρ^U_V restricts to the fiber over V, and ρ^V_W restricts to the fiber over W, so $\rho^V_W \circ \rho^U_V = \rho^U_W$.

We must also show that the restrictions are well defined.

- For $V \preccurlyeq U$, if $\langle b_i \rangle_{i \in I(U)} \in F(U)$, then by (G2) $\langle b_i \rangle_{i \in I(V)}$ satisfies $\mathcal{G}_V$, so ρ^U_V is well-defined. □

Theorem 2 ($\mathcal{G}$-sheaves are sheaves)**.** *Given a gluing condition $\mathcal{G}$ and an assignment of local data to the atomic regions of the underlying locale, the corresponding $\mathcal{G}$-sheaf satisfies the gluing condition G0.*

Proof. We must show that every gluable selection of patch candidates $\langle b_i \rangle_{i \in I(U)}$ glues to yield a unique section in $F(U)$. Assume that we have a compatible selection of patch candidates $\langle b_i \rangle_{i \in I(U)}$. Then:

- Existence: we assumed the patch candidates are compatible. By (G3) then, $\langle b_i \rangle_{i \in I(U)} \in F(U)$.
- Uniqueness: let $s = \langle b_i \rangle_{i \in I(U)} \in F(U)$. If another section $t = \langle b_i \rangle_{i \in I(U)} \in F(U)$ were glued from the same components, then $s = t$, since both restrict to the same supports. □

Throughout the rest of this paper, we will use $\mathcal{G}$-sheaves to model part-whole complexes, but that is only for simplicity of exposition. Any sheaf over a locale would do just as well.

2. Modeling Part-Whole Complexes as Sheaves

As noted in **??**, the central claim of this paper is that we can model part-whole complexes as sheaves over locales. In particular, the locale provides the abstract parts space of "regions" the pieces can occupy, the sheaf assigns actual pieces to those regions, and the gluing condition determines when pieces fuse.

We can thus define the core mereological concepts of part and whole in sheaf-theoretic terms. Regarding wholes, we can identify fusion with gluing: to say that some pieces fuse or form a "fusion" is just to say that they are glued together. Regarding parts, to say that a piece is a "part" is just to say that it is a part of a fusion. In other words, the parts of a fusion are just the pieces from which it is glued together.

Definition 17 (Fusions and parts)**.** *We say that a section $s \in F(U)$ is a fusion iff*

$$\mathcal{G}_U(s) = \text{true}.$$

Given $t \in F(V)$ and $s \in F(U)$ with $V \preccurlyeq U$ and $V \neq \bot$, we say t is a part of s, denoted $t \sqsubseteq s$, iff

$$\mathcal{G}_U(s) \quad \textit{and} \quad \rho^U_V(s) = t.$$

Remark 11. *$V \neq \bot$ because no parts can occupy $\bot$. The least region of the locale represents the combinatorial idea of no regions at all, and so it cannot be populated by any parts (hence in a $\mathcal{G}$-sheaf the sole section over $\bot$ is the singleton $\langle\rangle$).*

Sheaf theory thus provides a systematic framework with which to model a large variety of part-whole complexes in a "fusions-first" manner. In the rest of this section, we illustrate with examples.

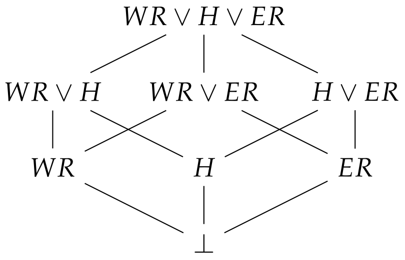
Example 14. *Consider a building with a west room, an east room, and a hallway between them. For simplicity, let us consider only the floors of the building (ignore walls, ceilings, and so on). The ambient locale is given by the presentation*

- $\mathbb{L} = \langle G, R\rangle = \langle\{WR, H, ER\}, \varnothing\rangle$

where

- *WR = west room*
- *H = hallway*
- *ER = east room*

As a Hasse diagram:



All of the generators are atomic, since none overlap (there are no meets among the generators):

- $Atoms(\mathbb{L}) = \{WR, H, ER\}$

Let us define a $\mathcal{G}$-sheaf F that models the flooring of this building. For data, let there be the following available flooring materials:

- $M = \{wood, tile, \ldots\}$

For a gluing condition, let us say that sections glue if they contain the same materials:

- $\mathcal{G}_U(\langle b_i\rangle_{i\in I(U)}) = \text{true}$ *iff $b_i = b_j$ for every $i, j \in I(U)$.*

- false *otherwise*

We must check that this is a legitimate gluing condition.

Proof. We must show that $\mathcal{G}$ satisfies the coherence conditions (G1)–(G3).

G1 *Local atomic data*. Trivial.

G2 *Downward stability*. We must show that if $\mathcal{G}_U(\langle b_i\rangle_{i\in I(U)})$ = true, then $\mathcal{G}_V(\rho^U_V(\langle b_i\rangle_{i\in I(U)}))$ = true for every $V \preccurlyeq U$. Assume the antecedent. Then $\rho^U_V(\langle b_i\rangle_{i\in I(U)}) = \langle b_i\rangle_{i\in I(V)}$.

- *Case 1*: if the length of $\langle b_i\rangle_{i\in I(V)} = 1$, it glues by (G1).
- *Case 2*: if the length of $\langle b_i\rangle_{i\in I(V)} \geqslant 2$, then by the assumption, $b_i = b_j$ for every $i, j \in I(V)$, so they glue.

G3 *Upward stability*. Given a selection of compatible patch candidates $\langle b_i\rangle_{i\in I(U)}$, we must show that if $\mathcal{G}_{U_i}(\langle b_i, b_j\rangle)$ = true for each $i, j \in I(U)$, then $\mathcal{G}_U(\langle b_i\rangle_{i\in I(U)})$ = true. Assume the antecedent. Since for every $i, j \in I(U)$, $b_i = b_j$ by the assumption, $\langle b_i\rangle_{i\in I(U)}$ glues. □

For the atomic regions, fix a choice of local data:

- $F(WR) = \{\langle wood\rangle\}$
- $F(H) = \{\langle wood\rangle\}$
- $F(ER) = \{\langle tile\rangle\}$

Extend compatible data to meets, of which there is only $\perp$, so:

- $F(\perp) = \{\langle\rangle\}$

Recursively extend data to joins via gluing:

- $F(WR \vee H) = \{\langle wood, wood\rangle\}$*, since* $F(WR) = F(H) = \{\langle wood\rangle\}$*, and wood = wood.*
- $F(WR \vee ER) = \varnothing$*, since* $F(WR) = \{\langle wood\rangle\}$*,* $F(ER) = \{\langle tile\rangle\}$*, and wood* $\neq$ *tile.*
- $F(H \vee ER) = \varnothing$*, since* $F(H) = \{\langle wood\rangle\}$*,* $F(ER) = \{\langle tile\rangle\}$*, wood* $\neq$ *tile.*
- $F(WR \vee H \vee ER) = \varnothing$*, since* $F(WR \vee H) = \{\langle wood, wood\rangle\}$*,* $F(H \vee ER) = F(WR \vee ER) = \varnothing$*, and wood* $\neq \varnothing$*.*

In this building, there are two maximal fusions:

- *The flooring of the west room and the hallway glue into one piece that covers both.*
- *The flooring that covers the east room is (trivially) glued into a single piece, namely itself.*

Thus, the flooring of this building is really a collection of two independent fusions: the wooden floor that covers the east room and hallway, and the tiled floor that covers the east room. That implies:

- *To separate the floors of the east room and hallway, you would have to use a saw to cut them, since they are fused. They are not merely sitting next to each other. Rather, they make up a single (fused) piece.*
- *By contrast, to separate the hallway and the east room, you would not need to cut them, since they are not fused. They simply happen to be sitting next to each other.*

The parts of the fusions are clear:

- *The wooden floor that covers the west room and the hallway has two parts: the wooden floor that covers the west room, and the wooden floor that covers the hallway.*
- *The tiled floor of the east room has no parts (in this locale), since it is not the fusion of other fusions.*

In the previous example, none of the atomic regions overlapped. The locale was discrete, and thus the sheaf was free to glue or not glue pieces as it saw fit. The story is different if there are overlaps in the locale itself. Overlaps in the locale require overlaps in the sheaf.

Example 15. *Consider the floor of a single room. Let us say that the regions of interest are its west half, its east half, and a six inch span where they overlap.*

The ambient locale of this kind of space can be given by the presentation

- $L = \langle G, R \rangle = \langle \{\bot, WH, O, EH\}, \{\bot \preccurlyeq O, O \preccurlyeq WH, O \preccurlyeq EH\} \rangle$

where

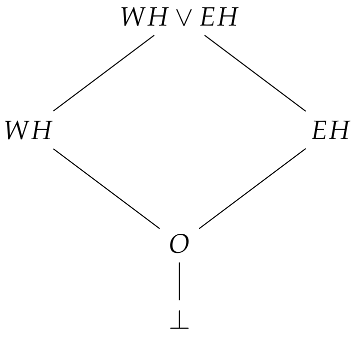
- WH = *west half*
- O = *overlap*
- EH = *east half*

The atomic sections of this locale are:

- WH
- EH
- $\bot$

In particular, O is not an atomic region, because it is the non-trivial overlap of WH and EH.

Here is the Hasse diagram:

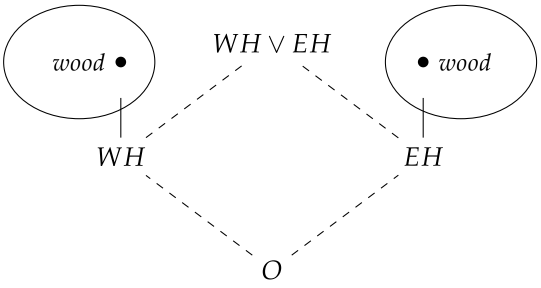


Let us define a $\mathcal{G}$-sheaf F that models the flooring of this room, using the same gluing condition from Example 14.

For the atomic regions, let us assign wood to both halves:

- $F(WH) = \{\langle wood \rangle\}$
- $F(EH) = \{\langle wood \rangle\}$

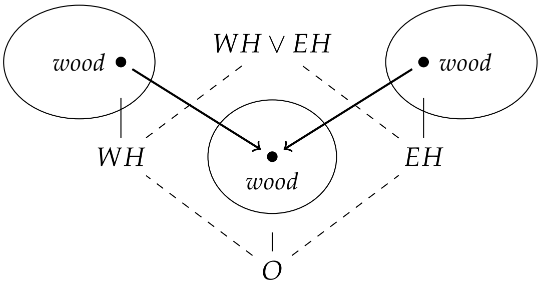
In a picture (omitting $\bot$ for simplicity):



For the meet (the overlap O), the two halves restrict to the same thing:

- $F(O) = \{\langle wood \rangle\}$

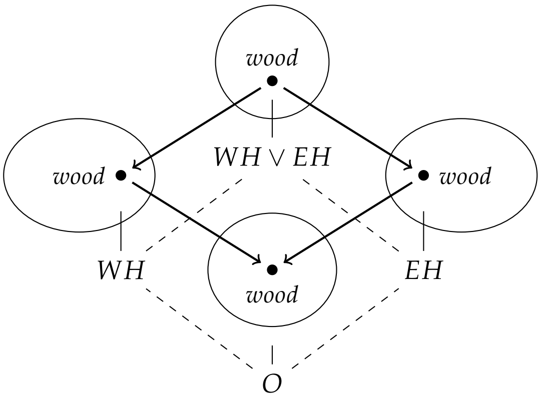
Thus:



For the join, the west and east halves glue, since they're made from the same flooring materials and agree on their overlap:

- $F(WH \vee EH) = \{\langle wood \rangle\}$

Thus:

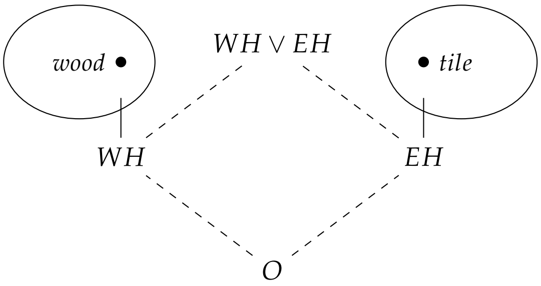


The maximal fusion is a single piece of wooden flooring that covers the whole room. Its parts are the west and east halves, and (transitively) their overlap. The west and east halves themselves have a shared part, the strip of overlap.

Example 16. *To illustrate a failed attempt to build a sheaf, let us take the locale and gluing condition from Example 15, but let's assign different flooring materials to the atomic regions:*

- $F(WH) = \{\langle wood \rangle\}$
- $F(EH) = \{\langle tile \rangle\}$

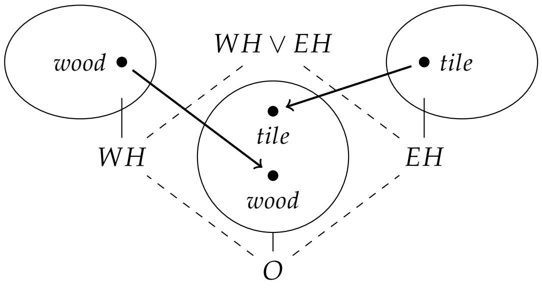
As a picture:



Next, we attempt to extend this data to meets, which requires that we restrict to the overlap, and then filter out anything that can't glue. Here, $F(WH)$ restricts to $\{\langle wood \rangle\}$, and $F(EH)$ restricts to $\{\langle tile \rangle\}$:

- $\rho^{WH}_{O}(\langle wood \rangle) = \langle wood \rangle$
- $\rho^{EH}_{O}(\langle tile \rangle) = \langle tile \rangle$

Thus:



However, these cannot glue, because they are not the same. We see here that the data of WH and EH disagree on the overlap. Hence, we are unable to construct a coherent sheaf. This illustrates how sheaf theory requires and manages coherent gluing at all levels. Because it requires that pieces glue together coherently at every level of "zoom," it prevents us from ever putting together an incoherent part-whole complex in the first place.

It is worth spelling the failure out explicitly. Since WH and EH disagree on their overlap, F cannot assign anything to O, so:

- $F(O) = \varnothing$

But that renders the restrictions ρ_O^{WH} and ρ_O^{EH} undefined, thereby severing our ability to zoom in and out. Thus, the system as a whole becomes incoherent.

Intuitively, this makes sense. If the western and eastern halves of a room were truly floored with different materials, then they would not overlap. There would be some sort of boundary between them where the one's materials end and the other's materials begin. But here, the ambient locale doesn't allow that possibility. In this *particular locale, the western and eastern halves* do *overlap, so the sheaf must assign pieces to the different regions coherently, i.e., it must assign pieces that agree on their overlap.*

The previous two examples were spatial. But parts come in non-spatial guises too, and sheaves can model them just as well.

Example 17. *Suppose we say that human society (under some description) consists of the mesh of a specified set of relationships between the people that participate in that society.*

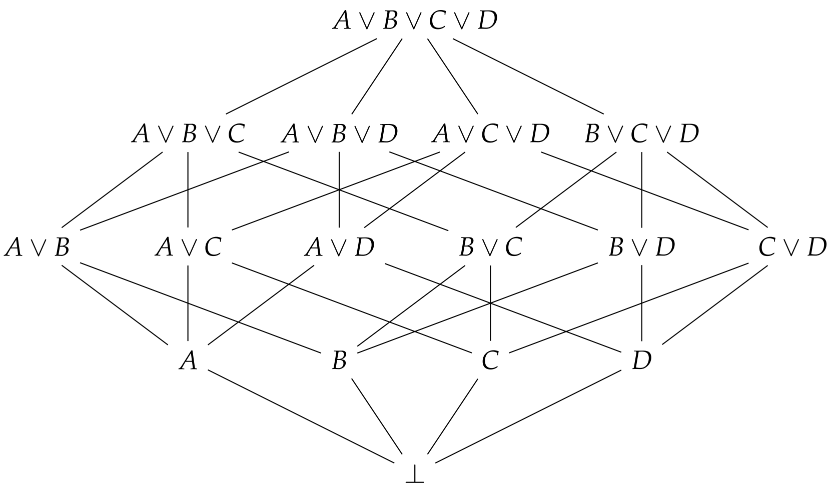
Let P be the population in question (a finite set of individual people), and let the regions of our locale be subsets of such individuals. Then the ambient locale is given by the presentation:

- $\mathbb{L} = \langle G, R \rangle = \langle P, \varnothing \rangle$

For concreteness, suppose:

- $P = \{$*A (Alice), B (Bob), C (Carol), D (Denny)* $\}$

Then the Hasse diagram is isomorphic to the powerset of P:



All of the generators are atomic, since there are no meets among the generators:

- *$Atoms(\mathbb{L}) = \{A, B, C, D\}$*

Let us define a $\mathcal{G}$-sheaf F that models the mesh of a selected set of relationships over P. To do that, let us first specify a set R that picks out the (binary, symmetric) relationships of interest:

- *$R = \{f$ (friends), m (married), $\ldots\}$*

For convenience, if $U, V \in P$, $r \in R$, and U and V stand in relationship r, we will write $r(U, V)$.

For a gluing condition, let us say that sections glue if they are connected by the same relations:

- $\mathcal{G}_U(\langle b_i \rangle_{i \in I(U)}) =$ true *iff for every $r \in R$, $r(U_i, U_j) \in b_i$ iff $r(U_j, U_i) \in b_j$, for every $i, j \in I(U)$*
- false *otherwise*

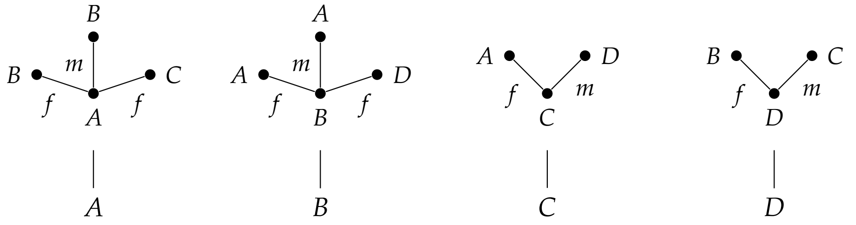
We must check that this is a legitimate gluing condition.

Proof. The proof is the same as before. □

For the atomic regions, let us fix a choice of local data by assigning to each person the relations they stand in, e.g.:

- $F(A) = \{\langle\{f(A, B), f(A, C), m(A, B)\}\rangle\}$
- $F(B) = \{\langle\{f(B, A), m(B, A), f(B, D)\}\rangle\}$
- $F(C) = \{\langle\{f(C, A), m(C, D)\}\rangle\}$
- $F(D) = \{\langle\{f(D, B), m(D, C)\}\rangle\}$

To visualize this data, we can picture each fiber as a mini-graph:



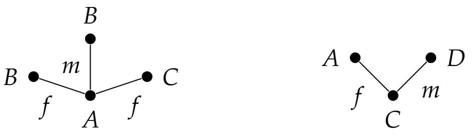
For example, in the fiber over A:

- *The f-labeled edge from A to B represents $f(A, B)$: A and B are friends.*
- *The m-labeled edge from A to B represents $m(A, B)$: A and B are married.*
- *The f-labeled edge from A to C represents $f(A, C)$: A and C are friends.*

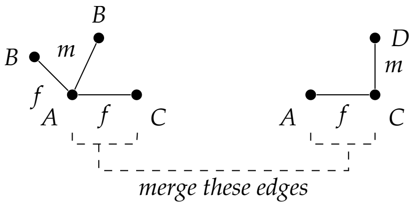
Next, we must extend compatible data to meets, of which there is only $\perp$, so:

- $F(\bot) = \{\langle\rangle\}$

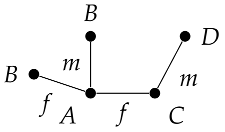
Finally, we must extend atomic data to binary joins via gluing. The gluing condition essentially says that mini-graphs can be glued along shared edges, provided that they share exactly the same edges. To see how this works, consider (for example) the mini-graphs over A and C:



Can these be glued? The answer is yes, because they share exactly one edge, namely the one labeled f. If you rotate the graphs sideways a bit, you can see how they can be merged along $f(A,C)$:



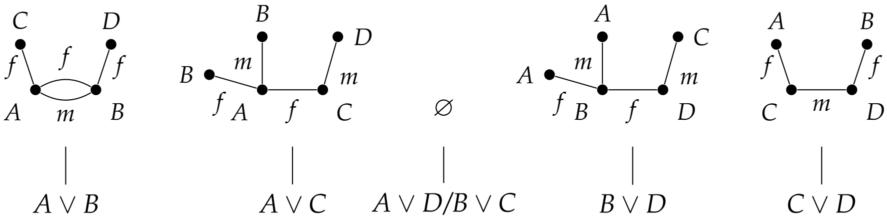
Merging along $f(A,C)$ yields the following glued graph:



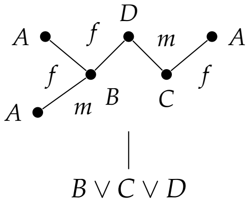
By gluing binary joins in this fashion, we get:

- $F(A \vee B) = \{\langle\{f(A,B), m(A,B), f(A,C)\}, \{f(B,A), m(B,A), f(B,D)\}\rangle\}$
- $F(A \vee C) = \{\langle\{f(B,A), m(B,A), f(B,D)\}\rangle, \{f(C,A), m(C,D)\}\}$
- $F(A \vee D) = \varnothing$
- $F(B \vee C) = \varnothing$
- $F(B \vee D) = \{\langle\{f(B,A), m(B,A), f(B,D)\}, \{f(D,B), m(D,C)\}\rangle\}$
- $F(C \vee D) = \{\langle\{f(C,A), m(C,D)\}, \{f(D,B), m(D,C)\}\rangle\}$

As pictures:



Having glued joins of two regions, we must next glue joins of three atomic regions. For instance, take $B \vee C \vee D$. We can glue $B \vee C$ trivially (because they share no edges), we can glue $C \vee D$ along their shared f-edge, and we can glue $B \vee D$ along their shared f-edge. That yields:



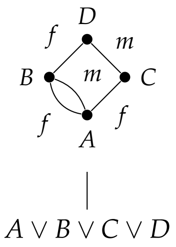
By gluing all joins of three atomic regions in this fashion, we get:

- $F(A \vee B \vee C) = \left\{ \begin{array}{ll} \langle & \{f(A,B), f(A,C), m(A,B)\}, \\ & \{f(B,A), m(B,A), f(B,D)\}, \\ & \{f(C,A), m(C,D)\} \quad \rangle \end{array} \right\}$
- $F(A \vee B \vee D) = \left\{ \begin{array}{ll} \langle & \{f(A,B), f(A,C), m(A,B)\}, \\ & \{f(B,A), m(B,A), f(B,D)\}, \\ & \{f(D,B), m(D,C)\} \quad \rangle \end{array} \right\}$
- $F(A \vee C \vee D) = \left\{ \begin{array}{ll} \langle & \{f(A,B), f(A,C), m(A,B)\}, \\ & \{f(C,A), m(C,D)\}, \\ & \{f(D,B), m(D,C)\} \quad \rangle \end{array} \right\}$
- $F(B \vee C \vee D) = \left\{ \begin{array}{ll} \langle & \{f(B,A), m(B,A), f(B,D)\}, \\ & \{f(C,A), m(C,D)\}, \\ & \{f(D,B), m(D,C)\} \quad \rangle \end{array} \right\}$

At the top-most join, gluing four regions, we get:

- $F(A \vee B \vee C \vee D) = \left\{ \begin{array}{ll} \langle & \{f(A,B), f(A,C), m(A,B)\}, \\ & \{f(B,A), m(B,A), f(B,D)\}, \\ & \{f(C,A), m(C,D)\}, \\ & \{f(D,B), m(D,C)\} \quad \rangle \end{array} \right\}$

As a picture:



The resulting sheaf yields a fused mesh of relationships over the population, which is glued together from smaller meshes over smaller subsets of the population.

- *Each atomic fiber is a part of the whole (human society), and its data encodes the internal (relational) structure of that part.*
- *Mereological overlap is then modeled by shared relationships: two parts overlap if their relational graphs intersect coherently.*
- *Failure to glue (as in $F(A \vee D) = \varnothing$ and $F(B \vee C) = \varnothing$) reflects mereological separation: the atomic regions in question cannot be fused because they are not related in this mesh.*

For another example, consider processes. A process (or more generally any sequence of events, states, etc.) can be seen as a part-whole complex too.

Example 18. *Imagine a scenario where something can do one of two things repeatedly: at each step, it can do one thing ("option a") or another thing ("option b"), and then repeat the choice again.*

To model this, fix a finite alphabet $\Sigma = \{a, b\}$, with "a" for "option a" and "b" for "option b." Then let Σ^* be the set of all finite sequences (words) over Σ, with ϵ denoting the empty sequence. For instance, the sequence aab represents the sequence of length 3 that picks "option a" first, then "option a" again, and then finally "option b."

Let us say that $\Sigma^{\leqslant n}$ is the set of all finite sequences less than length n, and let us say that $\Sigma^{=n}$ is the set of finite sequences of exactly length n. Hence:

- $\Sigma^{=0} = \{\epsilon\}$.
- $\Sigma^{=1} = \Sigma^{\leqslant 1} = \{\epsilon, a, b\}$.

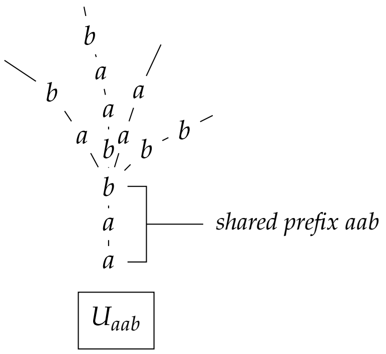
- $\Sigma^{\leqslant 2} = \{\epsilon, a, b, aa, bb, ab, ba\}$.
- $\Sigma^{=2} = \{aa, bb, ab, ba\}$.
- *Etc.*

Given sequences $w, v \in \Sigma^{\leqslant n}$ *with* $length(w) \leqslant length(v)$*, let us write* $w \subseteq v$ *to denote that* w *is a prefix of* v*, as in* $aab \subseteq aabc$.

Next, define a topology over $\Sigma^{\leqslant}$ *by setting the open sets to be sequences that share a prefix:*

- $U_w = \{v \in \Sigma^{\leqslant} \mid w \subseteq v\}$.

So U_w *consists of all sequences that continue* w*. For instance, if* $w = aab$*, then we might picture* U_w *as a kind of bouquet or bundle of sequences that are all bound at their shared stem (aab) but then branch out in different directions:*



We can form a locale from this topology. Let $\mathbb{L}$ *be the locale given by the presentation* $\langle G, R \rangle$*, where:*

- $G = \{U_w \mid w \in \Sigma^n\}$*, i.e., each open is a generator.*
- $R = \{U_w \preccurlyeq U_v \mid v \subseteq w\}$*, i.e., bouquets with longer prefixes are lower.*

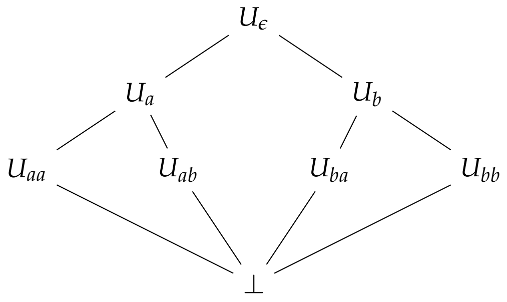
For example, given $\Sigma^{\leqslant 2}$*, we have the following generators:*

- $G = \{U_\epsilon, U_a, U_b, U_{aa}, U_{bb}, U_{ab}, U_{ba}\}$.

Here are some of the relations:

- $U_{aa} \preccurlyeq U_a$ *and* $U_{ab} \preccurlyeq U_a$*, since "a" is a prefix of aa and ab.*
- $U_{bb} \preccurlyeq U_b$ *and* $U_{ba} \preccurlyeq U_b$*, since "b" is a prefix of bb and ba.*
- *Every generator is lower than* U_ϵ*, since* ϵ *(the empty sequence) is a prefix of every sequence.*

The Hasse diagram looks like this:



Think of moving upwards in this locale as forgetting information about (or alternatively, as committing less to) the history of the sequence. For example, think of U_{ab} *as a region where we know that "a" happened first and then "b" happened, but think of* U_a *as a region where we know only that "a" happened first and we don't know what happened after that. The top element is* U_ϵ*, which means we don't know anything about the sequence of actions. The* $\perp$ *element indicates not that we know nothing, but that there is no sequence at all.*

Notice that implication moves upwards: U_{ab} *implies* U_a *because if I know (at* U_{ab}*) that "a" happened first and then "b" happened, then I certainly know that "a" happened first.*

Further, no generator is the non-trivial overlap of other generators, so every generator is an atomic region:

- $Atoms(\mathbb{L}) = G$

As with any locale, we can write each region canonically as the join of its atomic regions:

- $U_{I(U)} = \bigvee_{i \in I(U)} U_i$

But here, what this means is that we can canonically write each region as the join of its "most specified" prefixes. For instance, $I(U_a) = \{aa, ab\}$*, so:*

- $U_a = U_{I(U_a)} = \bigvee\{U_{aa}, U_{ab}\}$.

This makes sense. Since U_a *is a region where we know only that "a" happened first, it is the join of all maximal continuations that begin with "a."*

This particular locale is interesting because it models the "process space" of any 2-stage sequence that can make one of two choices at each stage. Let us now assign some actual processes to this ambient space, using a $\mathcal{G}$*-sheaf.*

Imagine a machine m *that can run multiple concurrent processes, all of whom share the same memory. For simplicity, let us suppose that the machine has two registers (*$R = \{r_1, r_2\}$*), each of which can hold one bit (*1 *or* 0*). So, at any point in time the machine's memory state* $S : \{0,1\} \times \{0,1\}$ *can be one of the following:*

- $S = \{\langle 0,0\rangle, \langle 1,0\rangle, \langle 0,1\rangle, \langle 1,1\rangle\}$*, with initial state* $s_0 = \langle 0,0\rangle$.

We can think of the concurrent processes of interest as a selection of programs that we want to run on the machine all at the same time. In terms of behavior, let us say that each program-run reads a word from its input stream, one character at a time, and in response to each character, it takes one of the following actions A*: it writes a value (*1 *or* 0*) to one of the registers, it writes (possibly distinct) values to both registers, or it does nothing and leaves the registers as they are:*

- $A = \{\{r1 \mapsto v\}, \{r2 \mapsto v\}, \{r1 \mapsto v, r2 \mapsto w\}, \varnothing\}$*, where* $v, w \in \{0,1\}$.

We can define a process (program trace) as a map from n-length words to n-length sequences of write actions, where we require that such maps agree on prefixes (since a process responding to ab and aa would do the same thing on the first a). This way, a program trace records for each input stream the sequence of write actions that result. For concreteness, here are two such traces:

- $f : \Sigma^{=2} \to A \times A$
  - $f(aa) = \langle\{r1 \mapsto 1\}, \{r1 \mapsto 0\}\}\rangle$
  - $f(ab) = \langle\{r1 \mapsto 1\}, \{r2 \mapsto 0\}\}\rangle$
  - $f(bb) = \langle\{\{r2 \mapsto 0\}, \{r2 \mapsto 1\}\}\rangle$
  - $f(ba) = \langle\{\{r2 \mapsto 0\}, \{r1 \mapsto 1\}\}\rangle$
- $g : \Sigma^{=2} \to A \times A$
  - $g(aa) = \langle\{r2 \mapsto 0\}, \{r2 \mapsto 0\}\}\rangle$
  - $g(ab) = \langle\{r2 \mapsto 0\}, \varnothing\}\rangle$
  - $g(bb) = \langle\{\{r1 \mapsto 1\}, \{r1 \mapsto 0\}\}\rangle$
  - $g(ba) = \langle\{\{r1 \mapsto 1\}, \{r1 \mapsto 1\}\}\rangle$

Let us now say that concurrent processes are compatible if they "play well" together, i.e., they share resources (memory) consistently. In particular, given two processes f *and* g*, let us say:*

- f *and* g *are compatible at stage* n *if they write to different registers.*
- f *and* g *are compatible at stage* n *if they write the same value to the same register.*

- *f and g conflict at stage n if they write different values to the same register.*

We can formalize this notion as a gluing condition that says a selection of patch candidates glue at U_w if they play well up to w. Fix a selection of programs $P = \{f, g, \ldots\}$ to run on the machine, then:

- *Given a selection of patch candidates $\langle (b_{i,p})_{p\in P}\rangle_{i\in I(U_w)}$ over a region U_w with trace length $|w|$, $\mathcal{G}_{U_w}(\langle b_i\rangle_{i\in I(U_w)}) = \text{true}$ iff the following condition holds. Write $b_{i,p}[m]$ to denote the write actions of process p in region i at stage m. Then, require that at each stage $m \leqslant |w|$ and every register $r \in R$, the set*

$$\{v \in \{0,1\} \mid \exists i \in I(U_w), p \in P \text{ where } r \mapsto v \in b_{i,p}[m]\}$$

has cardinality at most 1. In other words, two values are not written to the same register.
- $\mathcal{G}_{U_w}(\langle b_i\rangle_{i\in I(U_w)}) = \text{false}$ *otherwise.*

We must check that this is a legitimate gluing condition.

Proof. We must check that $\mathcal{G}_{U_w}$ is downwards and upwards stable.

- *Downwards stability*. Assume that $\mathcal{G}_{U_w}(\langle b_i\rangle_{i\in I(U_w)}) = \text{true}$. Then $\mathcal{G}_{U_i}(\langle b_i\rangle) = \text{true}$ for each $i \in I(U_w)$ since by $\mathcal{G}_{U_w}$, every b_i, b_j play well on their prefixes.
- *Upwards stability*. Assume $\mathcal{G}_{U_{\{i,j\}}}(\langle b_i, b_j\rangle) = \text{true}$ for all $i, j \in I(U_w)$. Then $\mathcal{G}_{U_w}(\langle b_i\rangle_{i\in I(U_w)}) = \text{true}$ since no i, j conflict on writes. □

With a gluing condition at hand, we can now define a $\mathcal{G}$-sheaf F. Let our selection of processes be $P = \{f, g, \ldots\}$. Then, we can fix the atomic data (omitting outer brackets to avoid clutter):

- $F(U_{aa}) = (f(aa), g(aa)) = (\langle\{r1 \mapsto 1\}, \{r1 \mapsto 0\}\rangle, \langle\{r2 \mapsto 0\}, \{r2 \mapsto 0\}\rangle)$.
- $F(U_{ab}) = (f(ab), g(ab)) = (\langle\{r1 \mapsto 1\}, \{r2 \mapsto 0\}\rangle, \langle\{r2 \mapsto 0\}, \varnothing\rangle)$.
- $F(U_{bb}) = (f(bb), g(bb)) = (\langle\{r2 \mapsto 0\}, \{r2 \mapsto 1\}\rangle, \langle\{r1 \mapsto 1\}, \{r1 \mapsto 0\}\rangle)$.
- $F(U_{ba}) = (f(ba), g(ba)) = (\langle\{r2 \mapsto 0\}, \{r1 \mapsto 1\}\rangle, \langle\{r1 \mapsto 1\}, \{r1 \mapsto 1\}\rangle)$.

There are no meets among the generators beyond $\perp$, so:

- $F(\perp) = \{\langle\rangle\}$.

Next, we must extend F to joins via gluing. So, for each U_w, we must assign:

- $F(U_w) = \{\langle (b_{i,p})_{p\in P}\rangle_{i\in I(U_w)} \mid \mathcal{G}_{U_w}(\langle (b_{i,p})_{p\in P}\rangle_{i\in I(U_w)}) = \text{true}\}$.

Let's compute $F(U_a) = F(U_{aa} \vee U_{ab})$. To determine if $(f(aa), g(aa))$ and $(f(ab), g(ab))$ glue, we need to check that they do not write conflicting values.

- *Stage 1 (at the shared prefix "a"): $f(aa)$ and $f(ab)$ write 1 to $r1$, while $g(aa)$ and $g(ab)$ write 0 to $r2$. Since f and g write to different registers, there is no conflict.*
- *Stage 2: $f(ab)$ and $g(aa)$ write the same value (namely, 0) to $r2$, $f(aa)$ writes 0 to $r1$, and $g(ab)$ does nothing, so there are no conflicts.*

Hence, $(f(aa), g(aa))$ and $(f(ab), g(ab))$ glue to form a unique section:

- $F(U_a) = (f(aa), g(aa)), (f(ab), g(ab))$

Now let's compute $F(U_b) = F(U_{bb} \vee U_{ba})$. To determine if $(f(bb), g(bb))$ and $(f(ba), g(ba))$ glue, we need to again check for conflicting writes:

- *Stage 1 (at the shared prefix "b"): $f(bb)$ and $f(ba)$ write 0 to $r2$, while $g(bb)$ and $g(ba)$ write 1 to $r1$, so there is no conflict.*
- *Stage 2: $f(bb)$ and $f(ba)$ write 1 to different registers, so they do not conflict with each other, while $f(ba)$ and $g(ba)$ write 1 to $r1$, so they do not conflict either. However, $g(bb)$ writes 0 to $r1$, which conflicts with $g(ba)$'s and $f(ba)$'s attempt to write 1 to the same register.*

Since we have a conflict, $(f(bb), g(bb))$ and $(f(ba), g(ba))$ fail to glue over U_b. Notice:

- *The processes f and g agree locally at U_a.*
- *By contrast, they disagree locally at U_b.*
- *There is no global section that glues together all of f and g's behavior at the top U_ϵ, thus f and g are not globally compatible processes.*

This sort of example illustrates how sheaves can model processes, concurrency, and resource conflicts. Here the processes were programs running on a simple machine, but they could just as easily be biological processes competing for resources, etc.

Whatever the concrete details may be, this example captures how local behavior can be integrated and extended over larger regions of the process space. One might naively think that the "parts" of such systems are the processes. But there is a different way to slice it: if you want to talk about the integrity of the "whole" of a concurrent system, you need to talk about how that involves coherent, integrated behavior that is functionally united locally across the various "regions" and "stages" of the system's evolution.

TODO:

- Add example: something over a continuous interval/timeline? E.g., maybe something over a timeline (the frame of opens taken from the usual topology of R)? Maybe we can define a gluing condition for a mass of clay over time that says pieces glue if they agree on overlaps, so that the whole lump of clay can have parts replaced over time but as a whole it never breaks into fragments? Maybe the "closure" is even a modality.
- Add example: Socrates and seated Socrates?
- Note Spivak et al's behavioral mereology is an example of a $\mathcal{G}$-sheaf (and check the details to make sure that's really true).

3. Modalities in the Sheaf-theoretic Setting

In the context of sheaves, modalities manifest as j-operators (also called local operators). A j-operator is a closure operator on the underlying locale.

Definition 18 (j-operators)**.** *Given a locale $\mathbb{L}$, a j-operator on $\mathbb{L}$ is a closure operator $j : \mathbb{L} \to \mathbb{L}$ satisfying the following conditions:*

(J1) Inflation. $U \preccurlyeq j(U)$.
(J2) Idempotence. $j(j(U)) = j(U)$.
(J3) Meet-preservation. $j(U \wedge V) = j(U) \wedge j(V)$.

A j-operator induces a j-sheaf.

Definition 19 (j-sheaves)**.** *Given a sheaf F over a locale $\mathbb{L}$ and a j-operator $j : \mathbb{L} \to \mathbb{L}$, the corresponding j-sheaf, denoted F_j, is given by:*

$$F_j = F(j(U)).$$

Remark 12. *In a sheaf, there are a variety of other modalities beyond the traditional alethic ones (necessity and possibility). Any closure operator qualifies as a modality of some description.*

Example 19. *From Example 17, recall the mesh of human relationships modeled by a $\mathcal{G}$-sheaf F defined over the presented locale $\mathbb{L} = \langle P := \{A, B, C, D\}, \varnothing\rangle$. Let us define a family of "reachability" modalities over this mesh.*

For each $r \in R$, write $\rightsquigarrow_r$ for the reflexive and transitive closure of r on the generators. Hence, $\rightsquigarrow_f$ is the transitive closure of friendship on the generators, and $\rightsquigarrow_m$ is the transitive closure on marriage.

Then for each $r \in R$, define j_r inductively:

- Base case. *For each generator $U \in G$, set j_r to the join of all other generators reachable via r:*

$$j_r(U) := \bigvee \{V \mid U \rightsquigarrow_r V\}$$

- Inductive step. *Extend to arbitrary joins $U_{I(U)}$:*

$$j_r(U_{I(U)}) := \bigvee_{i \in I(U)} j_r(U_i)$$

We need to check that this is a j-operator.

Proof. We check (J1)–(J3) from the definition.

Do the base case, then the inductive step.

□

Intuitively, this operator expands every region U to the largest region that is reachable from U by r. In other words, it expands each subset of society to the largest subset of society that is connected by r. Hence, $j_f(U)$ yields all those who are connected to U through a chain of friends, while $j_m(U)$ yields all those who are connected to U through a chain of marriage (which in a monogamous society will yield only married couples but in a polygamous society may be more revealing).

Applying j_f (for instance) to $\mathbb{L}$ yields the following:

- *$j_f(A) = A \vee B \vee C \vee D$, because $A \rightsquigarrow_f A$, $A \rightsquigarrow_f B$, $A \rightsquigarrow_f D$, and $A \rightsquigarrow_f C$.*
- *$j_f(B) = A \vee B \vee C \vee D$, because $B \rightsquigarrow_f B$, $B \rightsquigarrow_f D$, $B \rightsquigarrow_f A$, and $B \rightsquigarrow_f C$.*
- *Similar for $j_f(C)$ ad $j_f(D)$.*
- *$j_f(\bot) = \bot$.*

Hence, everyone in this mini-society is connected through friends (or friends-of-friends, etc.). Notice also that everyone is connected immediately*, i.e., at the first application of j_f.*

When it comes to marriage, the situation is different. Applying j_m yields:

- *$j_m(A) = A \vee B$, because $A \rightsquigarrow_m A$ and $A \rightsquigarrow_m B$.*
- *$j_m(B) = A \vee B$, because $B \rightsquigarrow_m B$ and $B \rightsquigarrow_m A$.*
- *$j_m(C) = C \vee D$, because $C \rightsquigarrow_m C$ and $C \rightsquigarrow_m D$.*
- *Similar for $j_m(D)$.*
- *$j_m(A \vee B) = A \vee B$, since A and B are already connected.*
- *$j_m(C \vee D) = C \vee D$, since C and D are already connected.*
- *$j_m(A \vee C) = A \vee B \vee C \vee D$, since from A, A can reach B (i.e., $A \rightsquigarrow_m B$) and from C, C can reach D (i.e., $C \rightsquigarrow_m D$).*
- *Similar for the rest.*

In contrast to the j_f modality, the j_m modality keeps the A, B component separate from the C, D component at all regions (sub-populations) that don't include a member of both couples, just as we would expect.

Now that we have defined j_f and j_m, we can construct a modal overlay for each that we can use to filter the original mesh:

- *Define the friendship mesh as F_f, filtered by j_f, i.e., set $F_f(U) := F(j_f(U))$.*
- *Define the marriage mesh as F_m, filtered by j_m, i.e., $F_m(U) := F(j_m(U))$.*

Example 20. *Recall the example of concurrent processes f and g from Example 18. We can define an* "already happened" *modality that captures what has definitely occurred so far.*

Definition 20 (Already-happened operator)**.** *Let j_H be given by:*

$$j_H(U_w) := \bigvee\{U_v \mid v \subseteq w\},$$

i.e., the join of all opens corresponding to prefixes of w (including w itself).

Intuitively, $j_H(U_w)$ is the region that remembers everything that has already happened along w. It is a closure operator that closes upwards by collecting all shorter prefixes.

We must check that this is a legitimate j-operator.

Proof. We check (J1)–(J3).

J1 *Inflation.* $U_w \preccurlyeq j_H(U_w)$ holds because U_w is included among the prefixes being joined.

J2 *Idempotence.* Applying j_H again adds no new prefixes, so $j_H(j_H(U_w)) = j_H(U_w)$.

J3 *Meet-preservation.* The meet of two regions corresponds to their longest shared prefix, whose prefixes are all of the prefixes collected by j_H. Hence, $j_H(U_w \wedge U_v) = j_H(U_w) \wedge j_H(U_v)$. □

Applying j_H to the generators of $\mathbb{L}$:

- *For U_{aa}: $j_H(U_{aa}) = U_\epsilon \vee U_a \vee U_{aa}$.*
- *For U_{ab}: $j_H(U_{ab}) = U_\epsilon \vee U_a \vee U_{ab}$.*
- *For U_{ba}: $j_H(U_{ba}) = U_\epsilon \vee U_b \vee U_{ba}$.*
- *For U_{bb}: $j_H(U_{bb}) = U_\epsilon \vee U_b \vee U_{bb}$.*
- *For U_a: $j_H(U_a) = U_\epsilon \vee U_a$.*
- *For U_b: $j_H(U_b) = U_\epsilon \vee U_b$.*
- *For U_ϵ: $j_H(U_\epsilon) = U_\epsilon$.*

Since j_H filters each region to everything that is already determined in that region, we can use it to define an overlay of F

$$F_H(U_w) := F(j_H(U_w)),$$

so that sections at U_w remember only what has happened along all prefixes of w.

Example 21. *Recall the example of concurrent processes f and g from Example 18. We can define a safety ("nothing bad happens") modality as a j-operator that identifies the largest safe extensions of a given region.*

Definition 21 (Safety operator)**.** *Let us say that a region U is safe if all processes in $F(U)$ play well together, i.e., if there are no write conflicts. Then let $j_S : \mathbb{L} \to \mathbb{L}$ be given by:*

$$j_S(U) := \begin{cases} \bigvee\{V \mid U \preccurlyeq V \text{ and } V \text{ is safe}\} & \text{if this join is non-empty} \\ U & \text{otherwise.} \end{cases}$$

Intuitively, $j_S(U)$ inflates U to the largest region that is guaranteed safe starting from U.

We must check that $j_S(U)$ is a legitimate j-operator.

Proof. We check (J1)–(J3).

J1 *Inflation.* By construction, $U \preccurlyeq j_S(U)$ whenever U has any safe parent regions, otherwise $j_S(U) = U$.

J2 *Idempotence*. Applying j_S more than once does not change the result, since appyling it once takes the join of all safe parents. Hence, $j_S(j_S(U)) = j_S(U)$.

J3 *Meet-preservation*. For any U and V, since $U \wedge V$ is U or V,

$$j_S(U \wedge V) = \bigvee\{W \mid U \wedge V \preccurlyeq W \text{ and } W \text{ safe}\} = j_S(U) \wedge j_S(V). \quad \square$$

Let's apply j_S to the generators of $\mathbb{L}$:

- *$j_S(U_{aa}) = U_a$ since its safe parent regions are U_{aa} and U_a.*
- *Similarly, $j_S(U_{ab}) = U_a$.*
- *$j_S(U_{ba}) = U_{ba}$ because the only safe parent of U_{ba} is U_{ba} itself.*
- *Similarly, $j_S(U_{bb}) = U_{bb}$.*

Now extend it to joins:

- *$j_S(U_a) = j_S(U_{aa}) \vee j_S(U_{ab}) = U_a \vee U_a = U_a$.*
- *$j_S(U_b) = U_b$ since U_b is unsafe (there are conflicts among its generators) and thus no further extension can be safe.*
- *$\perp$ is trivially fixed: $j_S(\perp) = \perp$.*

Notice:

- *Each generator U_w represents a part of a process's history.*
- *The operator j_S identifies the largest safe fusion containing U_w, i.e., the maximal extension of the part where processes play well together.*
- *If no safe extensions exist (as in U_b), then $j_S(U_b)$ doesn't get bigger, indicating that safety cannot be guaranteed any further beyond this part.*
- *Hence, j_S captures a mereological notion of integrity, showing which combinations of parts form consistent wholes and which do not.*

TODOs:

- Add example: A statue and the lump of clay?

4. Classical Mereological Notions in the Sheaf-theoretic Setting

In this section, we provide a discussion of what classical notions of mereology look like in the sheaf-theoretic setting.

- *Cambridge fusions*. Sheaves handle Cambridge fusions correctly.
- *Mere collections*. The collection of all dogs. Is that a "whole"? Well, we could build a sheaf whose atomic regions are filled with dogs, none of which glue. Then we have a collection of dogs, but no glued object. That matches exactly the intuition: yes, we have a "collection" (we built a sheaf for it, after all), but the internals of that sheaf reveal that it's *merely* a collection, i.e., that its parts are not glued.
- *Co-habitating fusions*. Sheaves allow multiple fusions to occupy the same locale, without being glued. For instance, in the sheaf of real-valued functions over real number line, there are many functions that glue together, and occupy the same locale.
- *Non-boolean algebra*. The parts space is Heyting, not Boolean. We're not saddled with such a strong complement operation. You can pick a locale that is Boolean if you need it, but this framework doesn't require it. In fact, the positive logic of a locale is "geometric logic."
- *Reflexivity, antisymmetry, and transitivity*. These are guaranteed. Locally, of course, you may not have transitivity. But globally, it's a theorem. [Check that.]
- *Distributivity*.

do the glued sections of a sheaf have to be distrubitive? Only inside what glues (since we glue pairwise, so every $i \vee j$ of the cover.

- *An empty element*. There is a need for a bottom element in the *algebra* of parts, but a sheaf need not contain any such thing. There is no need here to try and construct awkward mathematical structures that do algebra on parts but yet don't have a bottom element because our ontological intuitions tell us there can be no such thing. That confuses two issues: algebra and integrity. So here we separate those cleanly, and the algebra can do algebra while the sheaf can do integrity. [In a sheaf you CAN'T assign an empty element to bottom, for coherence, so the bottom element is special...need to say more about that and figure it out.]
- *Supplementation principles*. Sheaves don't constrain one way or another. [Is that really true? Maybe it's better to say that it doesn't force any supplementation principles, which might provide a reason to call into question whether supplementation is another one of those ideas that is about integrity of parts but has been confused with the algebra of parts.]
- *Ordering of parts*. Consider that "pit" and "tip" have the same parts but are different words. These differences can be handled by different sheaves over a 3-stage prefix-ordered locale as in the example of concurrent processes. Note that we retain extensionality.
- *Extensionality*. Classical mereology's notion of extensionality essentially flattens any structure and is thus overly aggressive. This is why extensionality is so controversial. The sheaf-theoretic perspective retains extensionality, but is much more nuanced. [Here too, I suspect that mereological discussions of extensionality have confused the algebra of parts and the integrity of wholes.]
- *Gunk and atoms*. You can model continuity and gunky parts if you so desire. You just need a sober space to do it.

  check that we can model continuity in the locale in this way.

  can you do continuity only in the sheaf data, without an underlying continuous decompositon in the locale? I would think that if you can't infinitely decompose into smaller opens around a point in the locale, you couldn't do such a thing in the sheaf data?
- *Priority of wholes*. The framework is agnostic as to whether you take an Aristotelian-Thomistic approach

  cite Aquinas, Arlig, and that guy who wrote that recent book defending the Aristotelian view

  .
- *The whole is greater than its parts*. The framework is agnostic as to whether you want to be a Scotist and say that the whole is something over and above its parts (cite Cross) or an Ockhamist who says it is not

  cite Normore, Arlig

  .

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Todo list