Appendix I

Proofs for Section 3.4.4

Here we present the technical details for Section 3.4.4. Define a set of display scalars as follows:

$$DS = \{red, green, blue, transparency, reflectivity, vector_x, vector_y, vector_z, contour_1, ..., contour_n, x, y, z, animation, selector_1, ..., selector_m\}$$

Also define a subset of display scalars $DOMDS = \{x, y, z, animation, selector_1, ..., selector_m\}$ and define $Y_{DOMDS} = \mathbf{X}\{I_d \mid d \in DOMDS\}$ and $Y = \mathbf{X}\{I_d \mid d \in DS\}$. Let $P_{DOMDS} : Y \to Y_{DOMDS}$ be the natural projection from Y onto Y_D (that is, if $a \in Y$ and $b = P_{DOMDS}(a)$, then for all $d \in DOMDS$, $b_d = a_d$). Then we can define $V_{display}$ as follows.

Def. $V_{display} = \{A \in V \mid \forall b, c \in MAX(A).\ P_{DOMDS}(b) = P_{DOMDS}(c) \Rightarrow b = c\}$. That is, if A is an object in $V_{display}$, then different tuples in A cannot have the same set of values for all display scalars in *DOMDS*.

In Prop. I.4 we will define conditions under which the displays of data objects are members of $V_{display}$. First, we prove three lemmas. Note that we use the notation a_d for the d component of a tuple $a \in \mathbf{X}\{I_d \mid d \in DS\}$.

Prop. I.1. Given a type $t \in T$ and $A \in D(F_t)$, then, for all tuples $a \in A$, $\forall d \in DS.\ (d \notin MAP_D(SC(t)) \Rightarrow a_d = \bot)$.

Proof. There is $B \in F_t$ such that $A = D(B)$. By Prop. F.12 for any $a \in A$ there is $b \in U$ such that $\downarrow a = D(\downarrow b)$. Since $\downarrow a \le A$, $\downarrow b \le B$ so $b \in B$. Furthermore, by Prop. F.12, if $a_d \neq \bot$ then there is $s \in S$ and $b_s \neq \bot$ such that $\downarrow(\bot,...,a_d,...,\bot) = D(\downarrow(\bot,...,b_s,...,\bot))$ and $d \in MAP_D(s)$. By Prop. D.1, $\forall s \in S.\ (b_s \neq \bot \Rightarrow s \in SC(t))$. Thus $a_d \neq \bot \Rightarrow d \in MAP_D(SC(t))$. ■

Prop. I.2. Given a tuple type $t = struct\{t_1;...;t_n\} \in T$, $A \in D(F_t)$ and $a = a_1 \vee ... \vee a_n \in A$, where $\forall i\ .\ a_i \in A_i \in D(F_{t_i})$, then $a \in MAX(A) \Leftrightarrow \forall i.\ a_i \in MAX(A_i)$.

Proof. Note that a and the a_i are tuples, and the *sup* of tuples is taken componentwise, so $\forall d \in DS.\ a_d = a_{1d} \vee ... \vee a_{nd}$. Also note that $i \neq j \Rightarrow SC(t_i) \cap SC(t_j) = \phi$, and, by Prop. F.9, $i \neq j \Rightarrow MAP_D(SC(t_i)) \cap MAP_D(SC(t_j)) = \phi$. If there is some i such that $a_i \notin MAX(A_i)$, then $\exists b_i \in A_i.\ a_i < b_i$ so $b = a_1 \vee ... \vee b_i \vee ... \vee a_n \in A$. Now, $a_i < b_i \Rightarrow \exists d \in DS.\ a_{id} < b_{id}$ and (since $j \neq i \Rightarrow a_{jd} = \bot = b_{jd}$) $a_d = a_{id}$ and $b_d = b_{id}$, so $a < b$. Thus $a \notin MAX(A)$. Conversely, if $a \notin MAX(A)$ then $\exists b \in A.\ a < b$ with $a = a_1 \vee ... \vee a_n$, $b = b_1 \vee ... \vee b_n$, and $\forall i.\ a_i, b_i \in A_i$. For some $d \in DS$, $a_d < b_d$. Thus $b_d > \bot$ so $\exists j.\ d \in MAP_D(SC(t_j))$, and so $a_d < b_d \Rightarrow a_j < b_j$ (since $a_d = a_{jd}$ and $b_d = b_{jd}$). Thus $a_j \notin MAX(A_j)$. ■

Prop. I.3. Given a tuple type $t = struct\{t_1;...;t_n\} \in T$, and given $B_i \in F_{t_i}$ and $A_i = D(B_i)$ for i=1,..,n, then:

(a) if $b_i \in B_i$ and $\downarrow a_i = \mathrm{D}(\downarrow b_i)$ for i=1,..,n, then $\downarrow(a_1 \vee ... \vee a_n) = D(\downarrow(b_1 \vee ... \vee b_n))$

(b) $A_i = \{a_i \mid \exists b_i \in B_i. \downarrow a_i = \mathrm{D}(\downarrow b_i)\}$

(c) $\bigvee\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in A_i\} = \{a_1 \vee ... \vee a_n \mid \forall i.\ a_i \in A_i\}$

Proof. First we prove (a). Note that the a_i and b_i are tuples. By Prop. D.1, $\forall i \neq j.\ \forall s \in S.\ (b_{is} = \bot \text{ or } b_{js} = \bot)$, so $(b_1 \vee ... \vee b_n)$ exists. Also, by Prop. D.1 and by Prop. F.12, $\forall d \in DS.\ d \notin MAP_D(SC(t_i)) \Rightarrow a_d = \bot$, and by Prop. F.9, $\forall i \neq j.\ MAP_D(\text{SC}(t_i)) \cap MAP_D(\text{SC}(t_j)) = \phi$, so $\forall i \neq j.\ \forall d \in DS.\ (a_{id} = \bot \text{ or } a_{jd} = \bot)$, and so $(a_1 \vee ... \vee a_n)$ exists. Given $\downarrow a_i = \text{D}(\downarrow b_i)$ then by Prop. F.12, the components of b_i determine the components of a_i. If $\downarrow x = D(\downarrow(b_1 \vee ... \vee b_n))$ then the components of $(b_1 \vee ... \vee b_n)$ determine the components of x. Since $\forall i \neq j.\ \forall s \in S.\ (b_{is} = \bot \text{ or } b_{js} = \bot)$, the components of $(b_1 \vee ... \vee b_n)$ are just the components of each of the b_i, so $x = (a_1 \vee ... \vee a_n)$, proving (a).

By Prop. F.12, for all $b_i \in B_i$ there is $a_i \in A_i = D(B_i)$ such that $\downarrow a_i = \text{D}(\downarrow b_i)$, so $A_i \supseteq \{a_i \mid \exists b_i \in B_i.\downarrow a_i = \text{D}(\downarrow b_i)\}$. Conversely, by Prop. F.12, for all $a_i \in A_i$ there is $b_i \in B_i$ such that $\downarrow a_i = \text{D}(\downarrow b_i)$, so $A_i \subseteq \{a_i \mid \exists b_i \in B_i.\downarrow a_i = \text{D}(\downarrow b_i)\}$. Together these prove (b).

Clearly, $\bigvee\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in A_i\} \supseteq \{a_1 \vee ... \vee a_n \mid \forall i.\ a_i \in A_i\}$. Pick $a \in \bigvee\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in A_i\}$. By Prop. C.10, there is a directed set $M \subseteq \bigcup\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in A_i\}$ such that $a = \bigvee M$. However, $\bigcup\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in A_i\} = \{c \mid (\forall i.\ \exists a_i \in A_i).\ c \leq (a_1 \vee ... \vee a_n)\}$. Now, for $c \leq (a_1 \vee ... \vee a_n)$, by Prop. C.9, $c = ((c \wedge a_1) \vee ... \vee (c \wedge a_n))$ where $(c \wedge a_i) \in A_i$, so $c \in \{a_1 \vee ... \vee a_n \mid a_i \in A_i\}$. Thus $M \subseteq \{a_1 \vee ... \vee a_n \mid a_i \in A_i\}$ such that $a = \bigvee M$. For each $m \in M$, let $m = (m_1 \vee ... \vee m_n)$ where $m_i \in A_i$. Then, since *sups* of tuples are taken componentwise and since $\forall i \neq j.\ \forall d \in DS.\ (m_{id} = \bot \text{ or } m_{jd} = \bot))$, $a = \bigvee M = \{(\bigvee m_1) \vee ... \vee (\bigvee m_n)) \mid m \in M\}$. However, $(\bigvee m_i) \in A_i$ since A_i is closed, so $a \in \{a_1 \vee ... \vee a_n \mid a_i \in A_i\}$. This proves (c). ■

Now we show that $MAX(A)$ is finite for data objects of types $t \in T$, and demonstrate conditions on t and D that ensure that displays of data objects of type t are in $V_{display}$.

Prop. I.4. If D is a display function, then for all types $t \in T$ and all $A \in D(F_t)$, $MAX(A)$ is finite. Furthermore, $MAP_D(DOM(t)) \subseteq DOMDS \Rightarrow D(F_t) \subseteq V_{display}$.

Proof. We will demonstrate both parts of this proposition by induction on the structure of t. Note that if t' is a subtype of t, then $MAP_D(DOM(t')) \subseteq MAP_D(DOM(t))$. Thus, if t satisfies the hypothesis of the second part, then its subtypes also satisfy the hypothesis of the second part.

Let $t \in S$ (note that $MAP_D(DOM(t)) = \phi \subseteq DOMDS$) and let $A \in D(F_t)$. Then, by the Theorem F.14, $\exists d \in MAP_D(t).\ A \in V_d$. Furthermore, $A \in V_d \Rightarrow \exists a \in I_d.\ A = \downarrow(\bot,...,a,...,\bot)$, so $MAX(A) = \{(\bot,...,a,...,\bot)\}$. $MAX(A)$ has a single member and is thus finite. Therefore $A \in V_{display}$ and thus $t \in S \Rightarrow D(F_t) \subseteq V_{display}$.

Let $t = struct\{t_1;...;t_n\} \in T$. Given $A \in D(F_t)$ there is $B \in F_t$ such that $A = D(B)$ and $\exists B_1 \in F_{t_1} ... \exists B_n \in F_{t_n}$. $B = \{(b_1 \vee ... \vee b_n) \mid \forall i.\ b_i \in B_i\}$. Also let $A_i = D(B_i)$. Then

$A = D(B) =$

$D(\bigvee\{\downarrow b \mid b \in B\}) =$ (by Prop. B.3)

$\bigvee\{D(\downarrow b) \mid b \in B\} =$

$\bigvee\{D(\downarrow(b_1 \vee ... \vee b_n)) \mid \forall i.b_i \in B_i\} =$ (by Prop. I.3 (a))

$\bigvee\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.\downarrow a_i = D(\downarrow b_i)\ \&\ b_i \in B_i\} =$ (apply Prop. I.3 (b) to each i)

$\bigvee\{\downarrow(a_1 \vee ... \vee a_n) \mid \forall i.a_i \in A_i\} =$ (by Prop. I.3 (c))

$\{(a_1 \vee ... \vee a_n) \mid \forall i.a_i \in A_i\}$

Thus $A \in D(F_t) \Rightarrow \exists A_1 \in D(F_{t_1})... \exists A_n \in D(F_{t_n}).\ A = \{(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in A_i\}$ and by Prop. I.2, $MAX(A) = \{(a_1 \vee ... \vee a_n) \mid \forall i.\ a_i \in MAX(A_i)\}$. By the inductive hypothesis, the $MAX(A_i)$ are finite, so $MAX(A)$ is finite. Now assume that $MAP_D(DOM(t)) \subseteq DOMDS$ but that $A \notin V_{display}$ (that is, assume that the second part of the proposition is not true). Then $\exists b, c \in MAX(A).\ P_{DOMDS}(b) = P_{DOMDS}(c)\ \&\ b \neq c$. Let $b = b_1 \vee ... \vee b_n$ and $c = c_1 \vee ... \vee c_n$ where $\forall i.\ b_i, c_i \in A_i$. The *sups* are taken componentwise for the tuples b and c, so for all $d \in DS$, $b_d = b_{1d} \vee ... \vee b_{nd}$ and $c_d = c_{1d} \vee ... \vee c_{nd}$. Now $P_{DOMDS}(b) = P_{DOMDS}(c) \Rightarrow \forall d \in DOMDS.\ b_d = c_d$. Pick $d \in DOMDS$, and we will show that $\forall i.\ b_{id} = c_{id}$. If $\exists i.\ d \in MAP_D(SC(t_i))$ then $\forall i' \neq i.\ d \notin MAP_D(SC(t_{i'}))$ and hence $\forall i' \neq i.\ b_{i'd} = \bot = c_{i'd}$ so that $b_{id} = b_d = c_d = c_{id}$, and hence $\forall i.\ b_{id} = c_{id}$. If $\forall i.\ d \notin MAP_D(SC(t_i))$ then $\forall i.\ b_{id} = \bot = c_{id}$. Either way, $P_{DOMDS}(b) = P_{DOMDS}(c)$ implies that $\forall d \in DOMDS.\ \forall i.\ b_{id} = c_{id}$ and so $\forall i.\ P_{DOMDS}(b_i) = P_{DOMDS}(c_i)$. On the other hand, $b \neq c \Rightarrow \exists e \in DS.\ b_e \neq c_e$. However, $e \notin MAP_D(SC(t_i)) \Rightarrow b_{ie} = \bot = c_{ie}$ and $\forall i.\ e \notin MAP_D(SC(t_i))$ would imply $b_e = \bot = c_e$. Thus $\exists j.\ e \in MAP_D(SC(t_j))$, and for this j, $b_{je} = b_e = c_e = c_{je}$ (since $b_{ie} = \bot = c_{ie}$ for $i \neq j$). And this implies that, for this j, $b_j \neq c_j$. However, by the inductive hypothesis, $b_j = c_j$, since we have already shown that $P_{DOMDS}(b_j) = P_{DOMDS}(c_j)$. Thus the assumption that $A \notin V_{display}$ has led to a contradiction, so $D(F_t) \subseteq V_{display}$.

Let $t = (array\ [w]\ of\ r) \in T$. Given $A \in D(F_t)$ there is $B \in F_t$ such that $A = D(B)$, and there is a finite set $G \in FIN(H_w)$ and a function $a \in (G \rightarrow H_r)$ such that

$$B = \{b_1 \vee b_2 \mid g \in G\ \&\ b_1 \in E_w(g)\ \&\ b_2 \in E_r(a(g))\} =$$

$$\bigcup\{\{b_1 \vee b_2 \mid b_1 \in E_w(g) \;\&\; b_2 \in E_r(a(g))\} \mid g \in G\}$$

Define $B_w(g) = E_w(g) \in F_w$, $B_r(g) = E_r(a(g)) \in F_r$, $A_w(g) = D(B_w(g)) \in D(F_w)$ and $A_r(g) = D(B_r(g)) \in D(F_r)$. Then

$$B = \bigcup\{\{b_1 \vee b_2 \mid b_1 \in B_w(g) \;\&\; b_2 \in B_r(g)\} \mid g \in G\}$$

This is a finite union of objects in $F_{struct\{w;\, r\}}$ for the tuple type *struct*$\{w; r\}$. Thus, since the union of a finte set of closed sets is the *sup* of those sets, and since D preserves *sups*,

$$A = D(B) = \bigcup\{D(\{b_1 \vee b_2 \mid b_1 \in B_w(g) \;\&\; b_2 \in B_r(g)\}) \mid g \in G\}$$

which, as shown in the tuple case of this proof, is equal to

$$\bigcup\{\{a_1 \vee a_2 \mid a_1 \in A_w(g) \;\&\; a_2 \in A_r(g)\} \mid g \in G\}$$

Recall that $MAX(A)$ is the set of maximal elements of A, so it is clear that if $A = A_1 \cup A_2$, then $MAX(A) \subseteq MAX(A_1) \cup MAX(A_2)$. Thus

$$MAX(A) \subseteq \bigcup\{MAX(\{a_1 \vee a_2 \mid a_1 \in A_w(g) \;\&\; a_2 \in A_r(g)\}) \mid g \in G\}$$

and so, by Prop. I.2,

$$MAX(A) \subseteq \bigcup\{\{a_1 \vee a_2 \mid a_1 \in MAX(A_w(g)) \;\&\; a_2 \in MAX(A_r(g))\} \mid g \in G\}$$

G is finite, and by the inductive hypothesis, $MAX(A_w(g))$ and $MAX(A_r(g))$ are finite, so $MAX(A)$ is finite.

Now assume that $MAP_D(DOM(t)) \subseteq DOMDS$. As shown for scalars, $MAX(A_w(g))$ has a single member, $MAX(A_w(g)) = \{a_1(g)\}$. Applying Prop. F.12, $A_w(g) = \downarrow a_1(g) = D(E_w(g)) = \mathrm{D}(\downarrow b_1(g))$ where $b_1(g) = (\perp,...,g,...,\perp)$. If $g \neq g'$, then $b_1(g) \neq b_1(g')$ and $a_1(g) \neq a_1(g')$. Also, given g, there is $d \in MAP_D(w)$ such that $a_1(g) = (\perp,...,a_{1d}(g),...,\perp)$. Since $w \in DOM(t)$, then $MAP_D(w) \subseteq DOMDS$ and $d \in DOMDS$. Thus $g \neq g' \Rightarrow a_1(g) \neq a_1(g') \Rightarrow P_{DOMDS}(a_1(g)) \neq P_{DOMDS}(a_1(g'))$.

Now pick $e, f \in MAX(A)$ and assume that $P_{DOMDS}(e) = P_{DOMDS}(f)$. Let $e = e_1 \vee e_2$ and $f = f_1 \vee f_2$ with $e_1 \in MAX(A_w(g_e))$, $f_1 \in MAX(A_w(g_f))$, $e_2 \in MAX(A_r(g_e))$ and $f_2 \in MAX(A_w(g_f))$. From what we have just seen, $g_e \neq g_f \Rightarrow P_{DOMDS}(e_1) \neq P_{DOMDS}(f_1)$. However, since $w \notin \mathrm{SC}(r)$, $MAP_D(w) \cap MAP_D(\mathrm{SC}(r)) = \phi$ so $P_{DOMDS}(e_1) \neq P_{DOMDS}(f_1) \Rightarrow P_{DOMDS}(e) \neq P_{DOMDS}(f)$. This contradicts our assumption, so we must have $g_e = g_f$ and, since $MAX(A_w(g))$ has a single member for each g, $e_1 = f_1$. Now $e_2, f_2 \in MAX(A_r(g_e))$ and $MAP_D(w) \cap MAP_D(\mathrm{SC}(r)) = \phi$ implies that $P_{DOMDS}(e) = P_{DOMDS}(f) \Rightarrow P_{DOMDS}(e_2) = P_{DOMDS}(f_2)$. By the inductive hypothesis, $A_r(g_e) \in V_{display}$, so $P_{DOMDS}(e_2) = P_{DOMDS}(f_2) \Rightarrow e_2 = f_2$. Thus $e = e_1 \vee e_2 = f_1 \vee f_2 = f$, establishing that $A \in V_{display}$ and that $D(F_t) \subseteq V_{display}$. ■

The next proposition shows that the auxiliary function D' provides a way to compute the maximal tuples of display objects.

Prop. I.5. If D is a display function, if D' is the auxiliary function defined in Appendix H, if $t \in T$ and if $A \in F_t$, then $MAX(D(A)) = \{D'(a) \mid a \in MAX(A)\}$

Proof. By Prop. H.5, $D(A) = \{D'(a) \mid a \in A\}$. By Prop. H.2, D' is an order embedding, so, given $a, b \in A$, $\neg(a < b) \Leftrightarrow \neg(D'(a) < D'(b))$. Thus $a \in MAX(A) \Leftrightarrow D'(a) \in MAX(D(A))$. ■

The inverse of the second part of Prop. I.4 is almost true. The next two propositions make this precise.

Prop. I.6. If D is a display function, if $t = (array\ [w]\ of\ r) \in T$, and if $\exists g_1, g_2 \in H_w. (g_1 \neq g_2\ \&\ D(\downarrow(\perp,...,g_1,...,\perp)) = \downarrow b_1 \in V_{d_1}\ \&$

$$D(\downarrow(\perp,...,g_2,...,\perp)) = \downarrow b_2 \in V_{d_2}\ \&\ d_1, d_2 \notin DOMDS),$$

then $\exists A \in D(F_t). A \notin V_{display}$.

Proof. Let $G = \{g_1, g_2\} \in FIN(H_w)$, pick $C \in H_r$, and define $f \in (G \to H_r)$ by $f(g_1) = C$ and $f(g_2) = C$. Pick $c \in E_r(C)$ such that $D(\downarrow c) = \downarrow a$ and $a \in MAX(D(E_r(C)))$. Then $(\perp,...,g_1,...,\perp) \vee c$ and $(\perp,...,g_1,...,\perp) \vee c$ are both members of $E_t(f) \in F_t$. Note that $D(\downarrow((\perp,...,g_1,...,\perp) \vee c)) = \downarrow(a \vee b_1)$ and $D(\downarrow((\perp,...,g_2,...,\perp) \vee c)) = \downarrow(a \vee b_2)$, so $a \vee b_1$ and $a \vee b_2$ are both members of $D(E_t(f))$. Clearly $b_1 \in MAX(D(\downarrow(\perp,...,g_1,...,\perp)))$ and $b_2 \in MAX(D(\downarrow(\perp,...,g_2,...,\perp)))$ (since b_1 and b_2 are maximal in $\downarrow b_1$ and $\downarrow b_2$). Furthermore, since $w \notin SC(r)$, $d_1 \notin MAP_D(SC(r))$ and $d_2 \notin MAP_D(SC(r))$, so $a \vee b_1$ and $a \vee b_2$ are members of $MAX(D(E_t(f)))$. For all $d \in DOMDS$, $b_{1d} = \perp$ and $b_{2d} = \perp$, so $P_{DOMDS}(a \vee b_1) = P_{DOMDS}(a \vee b_2)$. Since $w \notin SC(r)$, $d_1 \notin MAP_D(SC(r))$ and $d_2 \notin MAP_D(SC(r))$, so $a_{d_1} = \perp$ and $a_{d_2} = \perp$. However, $g_1 \neq g_2$ so $b_1 \neq b_2$ and hence $(a \vee b_1)_{d_1} \neq (a \vee b_2)_{d_1}$ and $(a \vee b_1)_{d_2} \neq (a \vee b_2)_{d_2}$ (d_1 and d_2 may or may not be the same). Thus $(a \vee b_1) \neq (a \vee b_2)$, so $D(E_t(f)) \notin V_{display}$. ■

Prop. I.7. If D is a display function, if $t \in T$, and if t has a sub-type t' such that $\exists A' \in D(F_{t'}).\ A' \notin V_{display}$, then $\exists A \in D(F_t).\ A \notin V_{display}$.

Proof. By an inductive argument, it is enough to prove this when t' is an immediate sub-type of t. First, let t be a tuple $t = struct\{t_1;...;t_n\}$ where $t' = t_k$. Let $A_k = A'$ and pick $a_k, a_k' \in MAX(A_k)$ such that $P_{DOMDS}(a_k) = P_{DOMDS}(a_k')$ and $a_k \neq a_k'$. For $i \neq k$, pick $A_i \in D(F_{t_i})$ and $a_i \in MAX(A_i)$. Then define $A = \{b_1 \vee ... \vee b_n \mid b_i \in A_i\} \in D(F_t)$. For $i \neq j$, $MAP_D(SC(t_i)) \cap MAP_D(SC(t_j)) = \phi$ so $a = a_1 \vee ... \vee a_k \vee ... \vee a_n \in MAX(A)$ and $a' = a_1 \vee ... \vee a_k' \vee ... \vee a_n \in MAX(A)$. Now $P_{DOMDS}(a_1 \vee ... \vee a_n) = P_{DOMDS}(a_1) \vee ... \vee P_{DOMDS}(a_n)$ and $P_{DOMDS}(a_k) = P_{DOMDS}(a_k')$ so $P_{DOMDS}(a_1 \vee ... \vee a_k \vee ... \vee a_n) = P_{DOMDS}(a_1 \vee ... \vee a_k' \vee ... \vee a_n)$. However, $a_k \neq a_k'$ so $a_1 \vee ... \vee a_k \vee ... \vee a_n \neq a_1 \vee ... \vee a_k' \vee ... \vee a_n$. Thus $A \notin V_{display}$.

Next, let t be an array $t = (array\ [w]\ of\ r)$. In the proof of Prop. I.4 we saw that $MAX(B')$ has only a single member for any $B' \in D(F_w)$, and hence $B' \in V_{display}$. Thus $t' = r$ and $A' \in D(F_r)$. Pick $G = \{g\} \in FIN(H_w)$, pick $b, c \in MAX(A')$ such that $P_{DOMDS}(b) = P_{DOMDS}(c)$ and $b \neq c$, and define $f \in (G \rightarrow H_r)$ by $f(g) = E_r^{-1}(D^{-1}(A'))$ ($A' \in D(F_r)$ implies that $D^{-1}(A')$ exists, and $D^{-1}(A') \in F_r$ implies that $E_r^{-1}(D^{-1}(A'))$ exists). If $D(\downarrow(\bot,...,g,...,\bot)) = \downarrow a$ then $a \in MAX(D(E_w(g)))$ and so $a \vee b$ and $a \vee c$ are members of $MAX(D(E_t(f)))$ (since $MAP_D(w) \cap MAP_D(SC(r)) = \phi$). However, $P_{DOMDS}(a \vee b) = P_{DOMDS}(a \vee c)$ but $a \vee b \neq a \vee c$. Thus $A \notin V_{display}$. ■

Bibliography

Avila, R., Taosong H., Lichan H., A. Kaufman, H. Pfister, C. Silva, L. Sobierakski, and S. Wang, 1994; VolVis: a diversified volume visualization system. Proc. IEEE Visualization '94, 31-38.

Bancroft, G. V., F. J. Merrit, T. C. Plessel, P. G. Kelaita, R. K. McCabe, and A. Globus, 1990; FAST: a multi-processed environment for visualization of computational fluid dynamics. Proc. IEEE Visualization '90, 14-27.

Bertin, J., 1983; Semiology of Graphics. W. J. Berg, Jr. University of Wisconsin Press.

Beshers, C., and S. Feiner, 1992; Automated design of virtual worlds for visualizing multivariate relations. Proc. Visualization '92, IEEE. 283-290.

Bier, E. A., M. C. Stone, K. Pier, W. Buxton, and T. Rose, 1994; Toolglass and magic lenses: the see-through interface. Proc. ACM Siggraph, 73-80.

Brittain, D. L., J. Aller, M. Wilson, S-L. C. Wang, 1990; Design of an end-user data visualization system. Proc. IEEE Visualization '90, 323-328.

Brodlie, K., A. Poon, H. Wright, L. Brankin, G. Banecki, and A. Gay, 1993; GRASPARC - a problem solving environment integrating computation and visualization. Proc. IEEE Visualization '93, 102-109.

Brown, M. H., and R. Sedgewick, 1984; A System for algorithm animation. Computer Graphics 18(3), 177-186.

Chen, M., S. J. Mountford, and A. Sellen, 1988; A study in interactive 3-D rotation using 2-D control devices. Computer Graphics 22(4), 121-129.

Corrie, B., and P. Mackerras, 1993; Data shaders. Proc. IEEE Visualization '93, 275-282.

Davey, B. A., and H. A. Priestly, 1990; Introduction to Lattices and Order. Cambridge University Press.

DeFanti, T. A., M. D. Brown, and B. H. McCormick, 1989; Visualization: expanding scientific and engineering research opportunities. IEEE Computer 22(8), 12-25.

Domik, G. O., and B. Gutkauf, 1994; User modeling for adaptive visualization systems. Proc. IEEE Visualization '94, 217-223.

Duff, T., 1992; Interval arithmetic and recursive subdivision for implicit functions and constructive solid geometry. Computer Graphics 26(2), 131-138.

Foley, J. D., and A. Van Dam, 1982; Fundamentals of Interactive Computer Graphics. Addison-Wesley.

Gierz, G., K. H. Hofmann, K. Keimal, J. D. Lawson, M. Mislove and D. Scott, 1980; A Compendium of Continuous Lattices. Springer-Verlag.

Globus, A., C. Levit, and T. Lasinski, 1991; A tool for visualizing the topology of three-dimensional vector fields. Proc. IEEE Visualization '91, 33-40.

Green, N., and M. Kass, 1994; Error-bounded antialiased rendering of complex environments. Proc. ACM Siggraph, 59-66.

Gunter, C. A., and D. S. Scott, 1990; Semantic domains. In the Handbook of Theoretical Computer Science, Vol. B., J. van Leeuwen ed., The MIT Press/Elsevier, 633-674.

Haber, R. B., B. Lucas and N. Collins, 1991; A data model for scientific visualization with provisions for regular and irregular grids. Proc. Visualization 91. IEEE. 298-305.

Haberli, P., 1988; ConMan: A visual programming language for interactive graphics. Computer Graphics 22(4), 103-111.

Haeberli, P., and K. Akeley, 1990; The accumulation buffer: hardware support for high-quality rendering. Computer Graphics 24(4), 309-318.

Haltiner, G. J., and R. T. Williams, 1980; Numerical prediction and dynamic meteorology, second edition. John Wiley & Sons, p. 40.

Hanrahan, P., and J. Lawson, 1990; A language for shading and lighting calculations. Computer Graphics 24(4), 289-298.

Hansen, C. D., and P. Hinker, 1992; Massively parallel isosurface extraction. Proc. IEEE Visualization '92, 77-81.

Helman, J. L., and L. Hesselink, 1990; Surface representations of two- and three-dimensional fluid flow topology. Proc. IEEE Visualization '90, 6-13.

Hibbard, W., 1986; Computer generated imagery for 4-D meteorological data. Bull. Amer. Met. Soc., 67, 1362-1369.

Hibbard, W., 1986; 4-D display of meteorological data. Proceedings, 1986 Workshop on Interactive 3D Graphics. Chapel Hill, Siggraph, 23-36.

Hibbard, W., and D. Santek, 1989; Interactivity is the key. Chapel Hill Workshop on Volume Visualization, University of North Carolina, Chapel Hill, 39-43.

Hibbard, W., and D. Santek, 1989; Visualizing large data sets in the earth sciences. IEEE Computer 22(8), 53-57.

Hibbard, W., L. Uccellini, D. Santek, and K. Brill, 1989; Application of the 4-D McIDAS to a model diagnostic study of the Presidents' Day cyclone. Bull. Amer. Met. Soc., 70(11), 1394-1403.

Hibbard, W., and D. Santek, 1990; The VIS-5D system for easy interactive visualization. Proc. Visualization '90, IEEE. 28-35.

Hibbard, W., D. Santek, and G. Tripoli, 1991; Interactive atmospheric data access via high speed networks. Computer Networks and ISDN Systems, 22, 103-109.

Hibbard, W., C. R. Dyer, and B. E. Paul, 1992; Display of scientific data structures for algorithm visualization. Proc. IEEE Visualization '92, 139-146.

Hibbard. W. L., C. R. Dyer, and B. E. Paul, 1994; A lattice model for data display. Proc. IEEE Visualization '94, 310-317.

Hibbard, W. L., B. E. Paul, A. L. Battaiola, D. A. Santek, M-F. Voidrot-Martinez and C. R. Dyer, 1994; Interactive Visualization of Computations in the Earth and Space Sciences. IEEE Computer 27(7), 65-72.

Hultquist, J. P. M., and E. L. Raible, 1992; SuperGlue: A programming environment for scientific visualization. Proc. Visualization '92, IEEE. 243-250.

Itoh, T., and K. Koyamada, 1994; Isosurface generation by using extrema graphs. Proc. IEEE Visualization '94, 77-83.

Kass, M., 1992; CONDOR: constraint-based dataflow. Computer Graphics 26(2), 321-330.

Kochevar, P., Z. Ahmed, J. Shade, and C. Sharp, 1993; Bridging the gap between visualization and data management: a simple visualization management system. Proc. IEEE Visualization '93, 94-101.

Lang, U., R. Lang, and R. Ruhle, 1991; Integration of visualization and scientific calculation in a software system. Proc. IEEE Visualization '91, 268-273.

Lee, J. P., and G. G. Grinstein, 1994; Database Issues for Data Visualization. Proc. of IEEE Visualization '93 Workshop. Springer-Verlag.

Levkowitz, H., 1991; Color icons: merging color and texture perception for integrated visualization of multiple parameters. Proc. IEEE Visualization '91, 164-170.

Lischinski, D., B. Smits and D. P. Greenberg, 1994; Bounds and error estimates for radiosity. Proc. ACM Siggraph, 67-74.

Lohse, J., H. Rueter, K. Biolsi, and N. Walker, 1990; Classifying visual knowledge representations: a foundation for visualization research. Proc. IEEE Visualization '90, 131-138.

Lorensen, W., and H. Cline, 1987; Marching cubes: a high-resolution 3D surface construction algorithm. Computer Graphics, 21(4), 163-170.

Lorenz, E. N., 1963; Deterministic nonperiodic flow. J. Atmos. Sci., 20, 130-141.

Lucas, B., G. D. Abrams, N. S. Collins, D. A. Epstein, D. L. Gresh, and K. P. McAuliffe, 1992; An architecture for a scientific visualization system. Proc. IEEE Visualization '92, 107-114.

Mackinlay, J., 1986; Automating the design of graphical presentations of relational information. ACM Transactions on Graphics, 5(2), 110-141.

Matveyev, S. V., 1994; Approximation of isosurface in the marching cube: ambiguity problem. Proc. IEEE Visualization '94, 288-292.

McConnell, C., and D. Lawton, 1988; IU software environments. Proc. IUW, 666-677.

McCormick, B.H., T.A. DeFanti and M.D. Brown, eds., 1987; Visualization in scientific computing. Computer Graphics, 21(6).

Montani, C., R. Scateni, and R. Scopigno, 1994; Discretized marching cubes. Proc. IEEE Visualization '94, 281-287.

Moore, R. E., 1966; Interval Analysis. Prentice Hall.

Nadas, T., and A. Fournier, 1987; GRAPE: An environment to build display processes. Computer Graphics 21(4), 103-111.

Nielson, G. M., and B. Hamann, 1991; The asymptotic decider: resolving the ambiguity in marching cubes. Proc. IEEE Visualization '91, 83-91.

Ning, P., and L. Hesselink, 1993; Fast volume rendering of compressed data. Proc. IEEE Visualization '93, 11-18.

Perlin, K., and D. Fox, 1993; Pad: an alternative approach to the computer interface. Proc. ACM Siggraph, 57-64.

Potmesil, M., and E. Hoffert, 1987; FRAMES: Software tools for modeling, animation and rendering of 3D scenes. Computer Graphics 21(4), 75-84.

Rabin, R. M., S. Stadler, P. J. Wetzel, D. J. Stensrud, and M. Gregory, 1990; Observed effects of landscape variability on convective clouds. Bull. Amer. Meteor. Soc., 71, 272-280.

Ranjan, V., and A. Fournier, 1994; Volume models for volumetric data. IEEE Computer, 27(7), 28-36.

Rasure, J., D. Argiro, T. Sauer, and C. Williams, 1990; A visual language and software development environment for image processing. International J. of Imaging Systems and Technology, Vol. 2, 183-199.

Read, R. L., D. S. Fussell and A. Silberschatz, 1993; Algorithms for the sandbag: an approach to imprecise set representation. Technical Report TR-93-12, Department of Computer Sciences, University of Texas at Austin.

Robertson, P. K., 1990; A methodology for scientific data visualization: choosing representations based on a natural scene paradigm. Proc. IEEE Visualization '90, 114-123.

Robertson, P. K., 1991; A methodology for choosing data representations. Computer Graphics and Applications, 11(3), 56-67.

Robertson, P. K., R. A. Earnshaw, D. Thalman, M. Grave, J. Gallup and E. M. De Jong, 1994; Research issues in the foundations of visualization. Computer Graphics and Applications 14(2), 73-76.

Rogowitz, B. E., and L. A. Treinish, 1993; An architecture for rule-based visualization. Proc. IEEE Visualization '93, 236-243.

Rolf, J., and J. Helman, 1994. IRIS Performer: a high performance multiprocessing toolkit for real-time 3D graphics. Proc. ACM Siggraph, 381-394.

Sanders, W. T., R. J. Edgar, M. Juda, W. L. Kraushaar, D. McCammon, S. L. Snowden, J. Zhang, M. A. Skinner, K. Jahoda, R. Kelley, A. Smalle, C. Stahle, and A. Szymkowiak, 1993; Preliminary results from the Diffuse X-ray Spectrometer. EUV, X-ray, and Gamma-ray Instrumentation for Astronomy IV. SPIE, Vol. 2006, 221-232.

Schmidt, D. A., 1986; Denotational Semantics. Wm.C.Brown.

Schroeder, W. J., W. E. Lorenson, G. D. Montanaro and C. R. Volpe, 1992; VISAGE: An object-oriented scientific visualization system. Proc. Visualization '92, IEEE. 219-226.

Scott, D. S., 1971; The lattice of flow diagrams. In Symposium on Semantics of Algorithmic Languages, E. Engler. ed. Springer-Verlag, 311-366.

Scott, D. S., 1976; Data types as lattices. Siam J. Comput., 5(3), 522-587.

Scott, D. S., 1982; Lectures on a mathematical theory of computation, in: M. Broy and G. Schmidt, eds., *Theoretical Foundations of Programming Methodology*, NATO Advanced Study Institutes Series (Reidel, Dordrecht, 1982) 145-292.

Segal, M., 1990; Using tolerances to guarantee valid polyhedral modeling results. Computer Graphics 24(4), 105-114.

Senay, H., and E. Ignatius, 1991; Compositional analysis and synthesis of scientific data visualization techniques. In Scientific Visualization of Physical Phenomena, N. M. Patrikalakis, ed. Springer-Verlag, 269-281.

Senay, H., and E. Ignatius, 1994; A knowledge-based system for visualization design. Computer Graphics and Applications, 14(6), 36-47.

Snyder, J. M., 1992; Interval Analysis for computer graphics. Computer Graphics 26(2), 121-130.

Springmeyer, R. R., M. M. Blattner, and N. L. Max, 1992; A characterization of the scientific data analysis process. Proc. IEEE Visualization '92, 235-242.

Treinish, L. A., 1991; SIGGRAPH '90 workshop report: data structure and access software for scientific visualization. Computer Graphics 25(2), 104-118.

Tuchman, A., D. Jablonowski, and G. Cybenko, 1991; Run-time visualization of program data. Proc. IEEE Visualization '91, 255-261.

Twiddy, R., J. Cavallo, and S. M. Shiri, 1994; Restorer: a visualization technique for handling missing data. Proc. IEEE Visualization '94, 212-216.

Upson, C., T. Faulhaber, Jr., D. Kamins, D. Laidlaw, D. Schlegel, J. Vroom, R. Gurwitz, A. van Dam, 1989; The application visualization system: A computational environment for scientific visualization. Computer Graphics and Applications, 9(4), 30-42.

Wehrend, S., and C. Lewis, 1990; A problem-oriented classification of visualization techniques. Proc. IEEE Visualization '90, 139-143.

Williams, C, J. Rasure, and C. Hansen, 1992; The state of the art of visual languages for visualization. Proc. IEEE Visualization '92, 202-209.

Wyvill, G., McPheeters, C., and B. Wyvill, 1986; Data structures for soft objects. Visual Computer 2(4), 227-234.