DESIGN MINING IN LEPUS3/CLASS-Z: SEARCH SPACE AND ABSTRACTION/CONCRETIZATION OPERATORS

Technical Report CSM-473, ISSN 1744-8050

Epameinondas Gasparis, Amnon H. Eden

Department of Computing and Electronic Systems, University of Essex
Colchester, Essex CO4 3SQ, United Kingdom

Abstract. LePUS3 is a specification and modelling language designed to capture the building blocks of O-O design at different levels of abstraction. We identify the set of LePUS3 specifications that agree with (are satisfied by) an O-O program (represented by a LePUS3 design model) as the search space for a host of *design mining* problems such as: reverse engineering, design recovery, design pattern detection, design pattern discovery. We show that this search space is a mathematical lattice (with relation to a particular program) and we demonstrate how it can be traversed using a set of abstraction and concretization operators.

Keywords: LePUS3, design mining

Conventions:
$\vdash$ denotes deducibility in classical logic.
$\models$ denotes satisfiability as defined in [Eden et. al 2007].
Given set $\mathcal{S}$, $|\mathcal{S}|$ stands for the size of $\mathcal{S}$.
LePUS3 constant terms:

- Lower case fixed-width characters such as $\texttt{x}$ are reserved for *0*-dimensional constant terms (see also Definition 1)
- Capitalized fixed-width characters such as $\texttt{Y}$ are reserved for *1*-dimensional constant terms (see also Definition 1)
- $\texttt{x}^d$ stands for a constant term of dimension d

$\underline{Relation}$ refers to a relation, and $Relation$ refers to a relation symbol.

1 Preliminary definitions

In this section we provide or adopt from [Eden et. al 2007], [Eden et. al 2007b] all the required definitions.

Definition 1: A **design model** for LePUS3 is a finite model-theoretic structure $\mathfrak{M}=\langle \mathbb{U}_*,\mathbb{R},\mathcal{I}\rangle$ such that:

- $\mathbb{U}_*$, called the **universe** of $\mathfrak{M}$, is a finite set of entities such that $\mathbb{U}_* \triangleq \mathbb{U}_0 \cup \mathbb{U}_1$ where:
  - $\mathbb{U}_0$ is a finite set of primitive entities that we call entities of dimension 0
  - $\mathbb{U}_1 \triangleq \mathcal{P}(\mathbb{U}_0)$. An entity in $\mathbb{U}_1$ is called an entity of dimension 1
- $\mathbb{R}$ is a set of relations, including:
  - the unary relations $\underline{Class}$, $\underline{Method}$, $\underline{Signature}$, $\underline{Inheritable}$ and $\underline{Abstract}$
  - the binary relations $\underline{Inherit}$, $\underline{Member}$, $\underline{Produce}$, $\underline{Call}$, $\underline{Forward}$, $\underline{Create}$, $\underline{Return}$, $\underline{Aggregate}$ and $\underline{SignatureOf}$
- $\mathcal{I}$ is an **interpretation[1]** function as follows:
  - if c is a constant term then $\mathcal{I}(\mathrm{c})$ is an entity in $\mathbb{U}_*$
  - if c and s are constant terms, and $\mathcal{I}(\mathrm{s})\otimes\mathcal{I}(\mathrm{c})$ is defined, then $\mathcal{I}(\mathrm{s}\otimes\mathrm{c})= \mathcal{I}(\mathrm{s})\otimes\mathcal{I}(\mathrm{c})$

  if t is in the domain of $\mathcal{I}$ then $\mathcal{I}(t)$ is the interpretation of t
- $\mathfrak{M}$ fixes the interpretation of higher dimensional (non 0-dimensional) constants

Definition 2: A LePUS3 **ground formula** is a formula in one of the following:

- a declaration in the form $t : \mathbb{CLASS}$ (or $\mathbb{SIGNATURE}$) which is shorthand for $Class(t)$ (or $Signature(t)$)
- a formula in the form $UnaryRelation(t)$ where t is a 0-dimensional term
- a formula in the form $BinaryRelation(t_1,t_2)$ where t_1, t_2 are 0-dimensional terms

For example, the schema presented in Table 1 contains 5 ground formulas.

Definition 3: A LePUS3 **predicate formula** is one of the following:

- a formula in the form $\textsc{All}(UnaryRelation, T)$ where $\textsc{All}$ is a predicate and T higher dimensional term

[1] To make sure that we ignore cases where different terms have the same interpretation we shall consider in this document $\mathcal{I}$ to be a bijective function.

- a formula in the form $P(BinaryRelation, T_1, T_2)$ where P is the $\textsc{Total}$ or $\textsc{Isomorphic}$ predicate and T_1, T_2 are higher dimensional terms

For example, the schema presented in Table 1 contains 1 predicate formula.

Table 1 – A Servlet example schema

Servlet
$\texttt{aServlet},\texttt{anotherServlet}, \texttt{HTTPServlet} : \mathbb{CLASS}$
$\texttt{JavaCollections} : \mathcal{P}(\mathbb{CLASS})$
$Inherit(\texttt{aServlet}, \texttt{HTTPServlet})$
$Inherit(\texttt{anotherServlet}, \texttt{HTTPServlet})$
$TOTAL(Member, \texttt{aServlet}, \texttt{JavaCollections})$

Definition 4: A LePUS3 **well-formed formula (wff)** is one of the following:

- a declaration in the form $T : \mathcal{P}(\mathbb{CLASS})$ (or $\mathcal{P}(\mathbb{SIGNATURE})$), which is a short-hand for $ALL(Class, T)$ (or $ALL(Signature, T)$)
- a ground formula
- a predicate formula

For example, the schema presented in Table 1 contains 7 wffs.

Definition 5: A **LePUS3 specification** is a finite set of LePUS3 wffs.

Definition 6: A ground formula is satisfied by design model $\mathfrak{M}$ under the following conditions:

- $\mathfrak{M} \models UnaryRelation(t)$ if and only if $\mathcal{I}(t) \in \underline{UnaryRelation}$
- $\mathfrak{M} \models BinaryRelation(t_1, t_2)$ if and only if one of the following conditions hold:
  - $\langle \mathcal{I}(t_1), \mathcal{I}(t_2) \rangle \in \underline{BinaryRelation}$
  - **Subtyping:** There exists some class of dimension 0 $\underline{\texttt{subcls}}$ in $\mathbb{U}_*$ such that $\langle \mathcal{I}(t_1), \underline{\texttt{subcls}} \rangle \in \underline{BinaryRelation}$ and $\langle \underline{\texttt{subcls}}, \mathcal{I}(t_2) \rangle \in \underline{Inherit}^+$

Definition 7: An $\textsc{All}$ predicate formula of the form $\textsc{All}(UnaryRelation, T)$ is satisfied by design model $\mathfrak{M}$ if and only if for each entity $\underline{e}$ in $\mathcal{I}(T_1)$: $\mathfrak{M} \models \underline{UnaryRelation}(\mathrm{e})$

Definition 8: A $\textsc{Total}$ predicate formula of the form $\textsc{Total}(BinaryRelation, T_1, T_2)$ is satisfied by design model $\mathfrak{M}$ if and only if for each entity $\underline{e}_1$ in $\mathcal{I}(T_1)$ that is not an abstract method, there exists some $\underline{e}_2$ entity in $\mathcal{I}(T_2)$ such that $\mathfrak{M} \vDash BinaryRelation(e_1, e_2)$

Definition 9: An $\textsc{Isomorphic}$ predicate formula in the form $\textsc{Isomorphic}(BinaryRelation, T_1, T_2)$ is satisfied by design model $\mathfrak{M}$ if and only if there exists pair $\langle \underline{e}_1, \underline{e}_2 \rangle$ where $\underline{e}_1 \in \mathcal{I}(T_1)$ and $\underline{e}_2 \in \mathcal{I}(T_2)$ such that:

- $\mathfrak{M} \vDash BinaryRelation(e_1, e_2)$ unless $\underline{e}_1$, $\underline{e}_2$ are abstract and
- $\mathfrak{M} \vDash \textsc{Isomorphic}(BinaryRelation, T_1 - e_1, T_2 - e_2)$ unless both $T_1 - e_1$ and $T_2 - e_2$ are empty

where $\mathcal{I}(T - e) = \mathcal{I}(T) - \mathcal{I}(e)$

2 Search Space

In this section we introduce LePUS3 *bottom* and *top specifications* with relation to a design model $\mathfrak{M}$ (that satisfies them). We establish the conditions under which a specification is *in normal form* and show that the *set of specifications* and *set of specifications in normal form* (with relation to a design model $\mathfrak{M}$ that satisfies them) are lattice stuctures.

Definition 10: Given specifications Φ, Ψ and design model $\mathfrak{M}$ we write $\Phi \vdash_{\mathfrak{M}} \Psi$ if and only if:

- $\Phi \vdash \Psi$ given $\mathfrak{M}$
- $\mathfrak{M} \vDash \Phi$ implies $\mathfrak{M} \vDash \Psi$

For example given the schema in Table 1, there is no way to prove `Servlet` $\vdash$ `Servlet2` using some syntactic proof theory and in the general case it would not be satisfied by any model for LePUS3. However, given a particular design model $\mathfrak{M}$ that satisfies both `Servlet` and `Servlet2` we can prove that `Servlet` $\vdash_{\mathfrak{M}}$ `Servlet2` if we consider that:

$Inherit(\texttt{aServlet}, \texttt{HTTPServlet}) \wedge Inherit(\texttt{anotherServlet}, \texttt{HTTPServlet}) \vdash_{\mathfrak{M}}$
$Hiearachy(\texttt{Servlets})$

As from that specific design model $\mathfrak{M}$ we know that:

$\mathcal{I}(\texttt{Servlets}) = \{\mathcal{I}(\texttt{aServlet}), \mathcal{I}(\texttt{anotherServlet}), \mathcal{I}(\texttt{HTTPServlet})\}$

Table 2 – Another Servlet example schema

Servlet2
Servlets : $\mathbb{HIERARCY}$
JavaCollections : $\mathcal{P}(\mathbb{CLASS})$

Definition 11: Given specifications Φ, Ψ and design model $\mathfrak{M}$ we say that Φ is equivalent to Ψ written as $\Phi \equiv_{\mathfrak{M}} \Psi$ if and only if $\Phi \vdash_{\mathfrak{M}} \Psi$ and $\Psi \vdash_{\mathfrak{M}} \Phi$.

Proposition 1: For any design model $\mathfrak{M}$, $\vdash_{\mathfrak{M}}$ is a partial order relation as $\vdash_{\mathfrak{M}}$ is:

- Reflexive, that is $\Psi \vdash_{\mathfrak{M}} \Psi$
- Anti-symmetric, that is if $\Psi \vdash_{\mathfrak{M}} \Phi$ and $\Phi \vdash_{\mathfrak{M}} \Psi$ then $\Psi \equiv_{\mathfrak{M}} \Phi$
- Transitive, that is if $\Psi \vdash_{\mathfrak{M}} \Phi$ and $\Phi \vdash_{\mathfrak{M}} \Omega$ then $\Psi \vdash_{\mathfrak{M}} \Omega$

Definition 12: $Spec(\mathfrak{M})$ is the set of all LePUS3 specifications that $\mathfrak{M}$ satisfies.

Corollary 1: $Spec(\mathfrak{M})$ is a partially ordered set with relation to $\vdash_{\mathfrak{M}}$.

Corollary 2: Given specifications Φ, Ψ if $\Phi \vdash_{\mathfrak{M}} \Psi$ then Φ, Ψ are in $Spec(\mathfrak{M})$.

Definition 13: A specification Φ is **in normal form** if and only if:

- Φ contains only ground formulas
- There exist no distinct ground formulas ψ, ϕ in Φ such that $\psi \vdash \phi$

2.1 Bottom and Top LePUS3 Specifications

Definition 14: A **bottom specification** $\perp_{\mathfrak{M}}$ with relation to a design model $\mathfrak{M}$ is a specification such that:

- $\perp_{\mathfrak{M}}$ is in normal form
- for any specification Φ, $\perp_{\mathfrak{M}} \vdash_{\mathfrak{M}} \Phi$

Definition 15: Let us call $Max_{\mathfrak{M}}$ a specification with relation to design model $\mathfrak{M}$ that is created by considering all tuples t in all relations in $\mathbb{R}$ such that:

$\forall t \in \bigcup_{\underline{\mathcal{R}} \in \mathbb{R}} \underline{\mathcal{R}}$

1) If $t \in \underline{Class}$ ($t \in \underline{Signature}$) then there exists exactly one *0*-dimensional constant term t of type $\mathbb{CLASS}$ ($\mathbb{SIGNATURE}$) in $Max_{\mathfrak{M}}$ such that $\mathcal{I}(\mathrm{t})$ is t

2) If $t \in \underline{Method}$ then there exists exactly one *0*-dimensional constant c of type $\mathbb{CLASS}$ and a *0*-dimension constant s of type $\mathbb{SIGNATURE}$ in $Max_{\mathfrak{M}}$ such that $(t,\mathcal{I}(\mathrm{s})) \in \underline{SignatureOf}$, $(t,\mathcal{I}(\mathrm{c})) \in \underline{Member}$ and $s \otimes c$ is a superimposition expression in at least one wff in $Max_{\mathfrak{M}}$

3) If $t \in \underline{Abstract}$ then there exists a *0*-dimensional constant term t in $Max_{\mathfrak{M}}$ such $\mathcal{I}(\mathrm{t})$ is t and $Abstract(\mathrm{t})$ is a wff in $Max_{\mathfrak{M}}$

4) If $t \in \underline{R}$, and $\underline{R}$ is one of the following: $\underline{Member}$, $\underline{Inherit}$, $\underline{Create}$, $\underline{Call}$, $\underline{Produce}$, $\underline{Return}$, $\underline{Forward}$ then t is a pair in the form (t_1,t_2) such that there exist *0*-dimensional constant terms t_1, t_2 in $Max_{\mathfrak{M}}$, $\mathcal{I}(\mathrm{t}_1)$ is t_1, $\mathcal{I}(\mathrm{t}_2)$ is t_2 and $R(\mathrm{t}_1,\mathrm{t}_2)$ is a wff in $Max_{\mathfrak{M}}$

Proposition 2: For any design model $\mathfrak{M}$, $Max_{\mathfrak{M}}$ is a bottom specification ($\perp_{\mathfrak{M}}$).

Proof

From Definition 15 we know that $Max_{\mathfrak{M}}$ contains all ground formulas that are satisfied by design model $\mathfrak{M}$. As it contains only ground formulas, it is in normal form (Definition 13). And as it contains all possible ground formulas that $\mathfrak{M}$ satisfies (Definition 6) it is a bottom specification.

■

Proposition 3: For any design model $\mathfrak{M}$, there is one bottom specification ($\perp_{\mathfrak{M}}$).

Proof

Since LePUS3 specification are sets of formulas, there is only one bottom specification that contains all and only ground formulas that $\mathfrak{M}$ satisfies (Definition 6).

■

Corollary 3: For any design model $\mathfrak{M}$ and respective bottom specification $\perp_{\mathfrak{M}}$, $\mathfrak{M} \vDash \perp_{\mathfrak{M}}$ (and $\perp_{\mathfrak{M}}$ is in $Spec(\mathfrak{M})$).

Definition 16: A **top specification** $\top_{\mathfrak{M}}$ with relation to a design model $\mathfrak{M}$ is a specification such that:

- $\top_{\mathfrak{M}}$ is in normal form
- for any specification Φ, $\Phi \vdash_{\mathfrak{M}} \top_{\mathfrak{M}}$

Definition 17: Let us call Min the specification which is the empty set: $Min=\{\}$.

Corollary 4: For any design model $\mathfrak{M}$, Min is a top specification ($\top_{\mathfrak{M}}$).

Corollary 5: For any design model $\mathfrak{M}$, there is one bottom specification $\top_{\mathfrak{M}}$.

Corollary 6: For any design model $\mathfrak{M} \vDash \top_{\mathfrak{M}}$ (and $\top_{\mathfrak{M}}$ is in $Spec(\mathfrak{M})$).

2.2 Normal Forms of LePUS3 Specifications

Given a specification in normal form, we examine its properties and establish when a specification is the normal form of another (Definition 18).

Corollary 7: For any specifications Φ, Φ' such that $\Phi' \subseteq \Phi$, if Φ is in normal form then Φ' is in normal form.

Proposition 4: Given specifications Φ, $\bot_{\mathfrak{M}}$ and design model $\mathfrak{M}$ such that $\mathfrak{M} \vDash \Phi$, Φ is in normal form if and only if $\Phi \subseteq \bot_{\mathfrak{M}}$.

Proof

If Φ is in normal form *then* $\Phi \subseteq \bot_{\mathfrak{M}}$.

As $\mathfrak{M} \vDash \Phi$, we know that there exists a specification $\bot_{\mathfrak{M}}$ (Definition 14) such that one of the following is true:

- $\Phi=\bot_{\mathfrak{M}}$ as $\bot_{\mathfrak{M}}$ is in normal form (Definition 14)
- $\Phi\neq\bot_{\mathfrak{M}}$. We know that $\bot_{\mathfrak{M}}$ contains all ground formulas that $\mathfrak{M}$ satisfies (Proposition 2). As $\bot_{\mathfrak{M}}$ is in normal form, for all ground formulas ψ in $\bot_{\mathfrak{M}}$ there does not exist ground formula ϕ in $\bot_{\mathfrak{M}}$ such that $\psi \vdash \phi$ (Definition 13). But also $\mathfrak{M} \vDash \Phi$, thus $\mathfrak{M}$ satisfies every ground formula in Φ (Definition 6), which means that every ground formula in Φ is also in $\bot_{\mathfrak{M}}$. That is $\Phi \subset \bot_{\mathfrak{M}}$

If $\Phi \subseteq \bot_{\mathfrak{M}}$ *then* Ψ is in normal form.

It follows from (Corollary 7) that Φ is in normal form as $\bot_{\mathfrak{M}}$ is in normal form (Proposition 2).

■

Proposition 5: Given specifications Φ, Ψ in normal form and design model $\mathfrak{M}$ that satisfies Φ, Ψ then $\Psi \subseteq \Phi$ if and only if $\Phi \vdash_{\mathfrak{M}} \Psi$.

Proof

If $\Psi \subseteq \Phi$ *then* $\Phi \vdash_{\mathfrak{M}} \Psi$.

Let $\Phi=\{\phi_1 \ldots \phi_n\}$ and $\Psi=\{\phi_x \ldots \phi_y\}$ with $1 \leq x \leq y \leq n$.

We know that $\mathfrak{M} \vDash \Phi$ which means that $\mathfrak{M}$ satisfies every formula in it.

Starting from the premise $\phi_1 \wedge \ldots \wedge \phi_n$ which is satisfied by $\mathfrak{M}$ and applying and-elimination we get:

$$\frac{\phi_1 \wedge \ldots \wedge \phi_n}{\phi_1 \wedge \ldots \wedge \phi_{n-1}} \wedge e_n$$

$$\frac{\phi_1 \wedge \ldots \wedge \phi_{n-1}}{\quad} \wedge e_{n-1}$$

…

$\phi_x \wedge \ldots \wedge \phi_y$ which is Ψ

If $\Phi \vdash_{\mathfrak{M}} \Psi$ *then* $\Psi \subseteq \Phi$.

Since $\mathfrak{M} \vDash \Phi$, there exists a bottom specification $\perp_{\mathfrak{M}}$ such that $\Phi \subseteq \perp_{\mathfrak{M}}$ and $\Psi \subseteq \perp_{\mathfrak{M}}$. (Proposition 4). From Definition 13 we know that for all ground formulas ψ in $\perp_{\mathfrak{M}}$ there does not exist specification ϕ such that $\psi \vdash \phi$. Thus if $\Phi \vdash_{\mathfrak{M}} \Psi$ it means that every wff in Ψ is also in Φ.

■

Corollary 8: There are no specifications Φ, Φ' in normal form and design model $\mathfrak{M}$ such that $\Phi' \subset \Phi$ and $\Phi' \vdash_{\mathfrak{M}} \Phi$.

Definition 18: Let Ψ, Φ be specifications and $\mathfrak{M}$ a design model such that $\mathfrak{M} \vDash \Phi$. We will say that Φ is **the normal form of** Ψ with relation to design model $\mathfrak{M}$ if and only if:

- Φ is in normal form
- $\Phi \vdash_{\mathfrak{M}} \Psi$
- There is no Φ' in normal form, such that $\Phi \vdash_{\mathfrak{M}} \Phi' \vdash_{\mathfrak{M}} \Psi$

Proposition 6: Given specifications Φ, Ψ, their respective normal forms Φ', Ψ' and design model $\mathfrak{M}$, if $\Phi \vdash_{\mathfrak{M}} \Psi$ then $\Phi' \vdash_{\mathfrak{M}} \Psi'$.

Proof

From our premise we know that $\Phi \vdash_{\mathfrak{M}} \Psi$ (*1*) and from Definition 18 we know that $\Phi' \vdash_{\mathfrak{M}} \Phi$ (*2*) and $\Psi' \vdash_{\mathfrak{M}} \Psi$ (*3*). Since Φ' and Ψ' are sets of ground formulas (Definition 13), one of the following is true:

- $\Phi' \cap \Psi' = \{\}$. In this case, from (*1*), (*2*) we can conclude: $\Phi' \vdash_{\mathfrak{M}} \Phi \vdash_{\mathfrak{M}} \Psi$ which means that $\Phi' \vdash_{\mathfrak{M}} \Psi$. Since Φ' and Ψ' are in normal form, given Definition 13, they should have at least one ground formula in common which is not true as it violates our assumption
- $\Phi' \cap \Psi' \neq \{\}$. In this case one of the following is true about Φ', Ψ' :
    - $\Phi' = \Psi'$. In this case $\Phi' \vdash_{\mathfrak{M}} \Psi'$ as relation $\vdash_{\mathfrak{M}}$ is reflexive (Proposition 1)
    - $\Phi' \subset \Psi'$. From Proposition 5 we know that $\Psi' \vdash_{\mathfrak{M}} \Phi'$. From (*2*) we can conclude $\Psi' \vdash_{\mathfrak{M}} \Phi' \vdash_{\mathfrak{M}} \Phi$ and from (*1*): $\Psi' \vdash_{\mathfrak{M}} \Phi' \vdash_{\mathfrak{M}} \Phi \vdash_{\mathfrak{M}} \Psi$. From Definition 18, we can conclude that Φ' would be the normal form of Ψ which is not true
    - $\Psi' \subset \Phi'$

We conclude that $\Psi' \subseteq \Phi'$ which given Proposition 5 means that $\Phi' \vdash_{\mathfrak{M}} \Psi'$.

■

Corollary 9: Given specifications Φ, Ψ, their respective normal forms Φ', Ψ' and design model $\mathfrak{M}$, if $\Phi \vdash_{\mathfrak{M}} \Psi$ then $\Psi' \subseteq \Phi'$.

2.3 Lattice Structures

Given the *set of specification in normal form* (Definition 19) (with relation to a design model $\mathfrak{M}$) and the *set of specifications* (with relation to a design model $\mathfrak{M}$), we show that each set is a mathematical lattice. For this reason we provide definitions of upper (lower) bound, supremum (infimum) and lattice that are based on the definitions found in [Burris & Sankappanavar 1981] and [Manzano 1999].

Definition 19: $Norm(\mathfrak{M})$ is the set of all LePUS3 specifications in normal form that $\mathfrak{M}$ satisfies.

Corollary 10: $Norm\ (\mathfrak{M})$ is a partially ordered set with relation to $\vdash_{\mathfrak{M}}$.

Corollary 11: $Norm\ (\mathfrak{M})$ is a subset of $Spec(\mathfrak{M})$.

Corollary 12: $\bot_{\mathfrak{M}}$ is in $Norm(\mathfrak{M})$.

Corollary 13: $\top_{\mathfrak{M}}$ is in $Norm(\mathfrak{M})$.

Definition 20: Let $\mathcal{A}$, $\mathcal{B}$ be sets such that $\mathcal{A} \subseteq \mathcal{B}$ and $\preceq$ a partial order relation on $\mathcal{B}$. An element b in $\mathcal{B}$ is an **upper bound** for $\mathcal{A}$ if for all a in $\mathcal{A}$ $a \preceq b$. An element b in $\mathcal{B}$ is a **lower bound** for $\mathcal{A}$ if for all a in $\mathcal{A}$ $b \preceq a$.

Definition 21: Let $\mathcal{A}$, $\mathcal{B}$ be sets such that $\mathcal{A} \subseteq \mathcal{B}$ and $\preceq$ a partial order relation on $\mathcal{B}$. An element b in $\mathcal{B}$, is the **least upper bound** of $\mathcal{A}$ if b is an upper bound of $\mathcal{A}$ and for all x that are upper bounds of $\mathcal{A}$ $b \preceq x$. If such b exists it is called the **supremum** of $\mathcal{A}$ or $Sup(\mathcal{A})$. An element b in $\mathcal{B}$ is the **greatest lower bound** of $\mathcal{A}$ if b is a lower bound of $\mathcal{A}$ and for all x that are lower bounds of $\mathcal{A}$ $x \preceq b$. If such b exists it is called the **infimum** of $\mathcal{A}$ or $Inf(\mathcal{A})$.

Definition 22: A partially ordered set $\mathcal{L}$ is a **lattice** if for all x, y in $\mathcal{L}$ both $Sup(\{x,y\})$ and $Inf(\{x,y\})$ exist (in $\mathcal{L}$).

Proposition 7: $\langle Norm(\mathfrak{M}), \vdash_{\mathfrak{M}} \rangle$ is a lattice.

Proof

For all specifications Ψ, Φ in $Norm(\mathfrak{M})$, $\{\Psi, \Phi\}$ is a subset of $Norm(\mathfrak{M})$. We know that $Norm(\mathfrak{M})$ is a partially ordered set (Corollary 10). Let us assume that $Inf(\{\Psi,\Phi\})=\Gamma$ exists and is in $Norm(\mathfrak{M})$.

If Γ is a lower bound (Definition 20) then: $\Gamma \vdash_{\mathfrak{M}} \Psi$ (*1*) and $\Gamma \vdash_{\mathfrak{M}} \Phi$ (*2*) for all Ψ, Φ.
Since Ψ, Φ and Γ are in normal form:
From Proposition 5 and (*1*) we know that: $\Psi \subseteq \Gamma$ (*3*)
From Proposition 5 and (*2*) we know that: $\Phi \subseteq \Gamma$ (*4*)
In order for Γ to be the greatest lower bound (Definition 21) given (*3*), (*4*) it needs to be $\Gamma = \Phi \cup \Psi$.

But $\Phi \cup \Psi$ is in normal form (Definition 13), as Φ, Ψ are and $\Phi \cup \Psi$ is a subset of $\bot_{\mathfrak{M}}$ (Proposition 4). Since there is exactly one subset of $\bot_{\mathfrak{M}}$ that contains all and only ground formulas in $\Phi \cup \Psi$ then Γ exists and is in $Norm(\mathfrak{M})$.

Symmetrically we can show that for any two specifications in $Norm(\mathfrak{M})$, $Sup(Norm(\mathfrak{M}))$ is$\top_{\mathfrak{M}}$.

■

Proposition 8: $\langle Spec(\mathfrak{M}),\vdash_{\mathfrak{M}}\rangle$ is a lattice.

Proof

For all specifications Ψ, Φ in $Spec(\mathfrak{M})$, {Ψ, Φ} is a subset of $Spec(\mathfrak{M})$. $Spec(\mathfrak{M})$ is a partially ordered set (Corollary 10). Let us assume that $Inf(\{\Psi,\Phi\})=\Gamma$ exists and is in $Spec(\mathfrak{M})$.

If Γ is a lower bound (Definition 20) then $\Gamma \vdash_{\mathfrak{M}} \Psi$ (*1*) and $\Gamma \vdash_{\mathfrak{M}} \Phi$ (*2*).
Given Corollary 2, if such Γ exists it will be in $Spec(\mathfrak{M})$.
If Γ is the least upper bound (Definition 21) there should not exist another upper bound Δ such that $\Gamma \vdash_{\mathfrak{M}} \Delta \vdash_{\mathfrak{M}} \Psi$ (*3*) and $\Gamma \vdash_{\mathfrak{M}} \Delta \vdash_{\mathfrak{M}} \Phi$ (*4*).

Let Γ', Δ', Φ', Ψ' be the normal forms of Γ, Δ, Φ, Ψ respectively.
Given Definition 18 if Γ exists, then there also exists specification Γ' in normal form such that $\Gamma' \vdash_{\mathfrak{M}} \Gamma$.

If both Γ' and Δ' exist:
From Proposition 6 and (*3*) we know that $\Gamma' \vdash_{\mathfrak{M}} \Delta' \vdash_{\mathfrak{M}} \Psi'$ (*5*)
From Proposition 6 and (*4*) we know that $\Gamma' \vdash_{\mathfrak{M}} \Delta' \vdash_{\mathfrak{M}} \Phi'$ (*6*)
From Corollary 9 and (*5*) we know that $\Psi' \subseteq \Delta' \subseteq \Gamma'$ (*7*)
From Corollary 9 and (*6*) we know that $\Phi' \subseteq \Delta' \subseteq \Gamma'$ (*8*)
Therefore, to prove that Γ is $Inf(\{\Psi,\Phi\})$ it is enough to show that Γ' exists and Δ' does not (unless $\Gamma' = \Delta'$).

From Proposition 6 and (*1*) we know that: $\Gamma' \vdash_{\mathfrak{M}} \Psi'$ (*9*)
From Proposition 6 and (*2*) we know that: $\Gamma' \vdash_{\mathfrak{M}} \Phi'$ (*10*)
From Corollary 9 and (*9*) we know that: $\Psi' \subseteq \Gamma'$ (*11*)
From Corollary 9 and (*10*) we know that: $\Phi' \subseteq \Gamma'$ (*12*)
From (*11*), (*12*) we conclude that Γ' should be: $\Gamma'=\Phi' \cup \Psi'$ so that there does not exist Δ' such that (*7*), (*8*) are true.

But there is exactly one subset of $\bot_{\mathfrak{M}}$ that contains all and only ground formulas in $\Phi' \cup \Psi'$, therefore Γ' exists and is in $Spec(\mathfrak{M})$ and so does Γ.

Symmetrically we can show that any two specifications in $Norm(\mathfrak{M})$, $Sup(Norm(\mathfrak{M}))$ is$\top_{\mathfrak{M}}$.

■

3 Operators

Given a design model $\mathfrak{M}$ and the set of specifications $Spec(\mathfrak{M})$ that $\mathfrak{M}$ satisfies, which is a lattice structure, we show how it is possible to traverse it by making *steps* (Definition 25) from one specification (node in the lattice) to another. Each step is performed by the application of an *operator* (Definition 26). Operators are divided into two sets: the *abstraction* and the *concretization* operators and are outlined in Table 3.

Definition 23: Let $\mathit{S}{\scriptstyle PEC}$ be the set of all LePUS3 specifications.

Definition 24: **Verbosity** of a specification Ψ written as $Verbosity(\Psi)$ is a function

$$Verbosity : \mathit{S}{\scriptstyle PEC} \rightarrow \mathbb{N}$$

such that values in its range calculate as the sum of the number of constant terms in Ψ and the number of wffs in Ψ.

Definition 25: Given $\mathfrak{M}$, we say that the transition from specification Ψ to Φ is an **abstraction step**, if the following conditions hold:

- $\mathfrak{M} \models \Psi$
- $\Psi \vdash_{\mathfrak{M}} \Phi$
- $Verbosity(\Psi) \geq Verbosity(\Phi)$

Remark: The transition from Φ to Ψ would be a **concretization step.**

Corollary 14: The transition from Ψ to Φ is an abstraction step if and only if the normal forms $\Psi' \subseteq \Phi'$.

Table 3a – Abstraction operators		**Table 3b – Concretization operators**	
Aggregation	... ⇒	Enumeration	... ⇐
Union	... ⇒	Partition	... ⇐
Hierarchy to Set	⇒	Set to Hierarchy	⇐
Collapse to Hierarchy	Inherit ⇒	Hierarchy Expansion	Inherit ⇐
Hierarchies Union	... ⇒	Partition to Hierarchies	... ⇐
To Top	$\top_{\mathfrak{M}}$	To Bottom	$\bot_{\mathfrak{M}}$
Elimination	⇒	Introduction	⇐

Definition 26: An **operator** $\mathcal{O}(\{t_1 \ldots t_n\},\Psi)$ takes a set of constant terms $\{t_1 \ldots t_n\}$ and specification Ψ, and produces $(\{t_1' \ldots t_m'\},\Phi)$ that is: a set of constant terms $\{t_1' \ldots t_m'\}$ and specification Φ, such that the following conditions hold:

- All $t_1, \ldots t_n$, are in Ψ
- All $t_1', \ldots t_n'$ are in Φ

All conditions in

- Definition 25 hold.

The set of operators is symmetric. If $\mathcal{O}$ is an abstraction operator that makes a transition from Ψ to Φ then there exists a concretization operator $\mathcal{O}'$ that makes a transition from Φ to Ψ and vice versa.

3.1 Concretization Operators

3.1.1 Enumeration

$$(\{\mathtt{T}\}, \Psi) \rightarrow (\{\mathtt{t}_1 \ldots \mathtt{t}_n\}, \Phi)$$

Pre-conditions:

- $\mathtt{T}$ is a term of type $\mathbb{CLASS}$ or $\mathbb{SIGNATURE}$

Post-conditions:

- Terms $\mathtt{t}_1 \ldots \mathtt{t}_n$ are all of the same type as $\mathtt{T}$ in Φ
- $\mathcal{I}(\mathtt{T}) = \{\mathcal{I}(\mathtt{t}_1) \ldots \mathcal{I}(\mathtt{t}_n)\}$

3.1.2 Partition

$$(\{\mathtt{T}\}, \Psi) \rightarrow (\{\mathtt{T}_1 \ldots \mathtt{T}_n\}, \Phi)$$

Pre-conditions:

- $\mathtt{T}$ is a term of type $\mathbb{CLASS}$ or $\mathbb{SIGNATURE}$
- $|\mathcal{I}(\mathtt{T})| \geq 2$

Post-conditions:

- Terms $\mathtt{T}_1 \ldots \mathtt{T}_n$ all of the same type as $\mathtt{T}$ in Φ
- $\mathcal{I}(\mathtt{T}) = \mathcal{I}(\mathtt{T}_1) \cup \ldots \cup \mathcal{I}(\mathtt{T}_n)$
- For at least $n-1$ terms $\mathtt{T}_i$, $1 \leq i \leq n$ introduced there exists at least one formula of the following forms with that term that is satisfied by $\mathfrak{M}$:
    - $\textit{Total}(\textit{BinaryRelation}, \mathtt{x}^d, \mathtt{T}_i)$
    - $\textit{Isomorphic}(\textit{BinaryRelation}, \mathtt{x}^d, \mathtt{T}_i)$
    - $\textit{Total}(\textit{BinaryRelation}, \mathtt{T}_i, \mathtt{x}^d)$
    - $\textit{Isomorphic}(\textit{BinaryRelation}, \mathtt{T}_i, \mathtt{x}^d)$
    - $\textit{All}(\textit{BinaryRelation}, \mathtt{T}_i)$
    - $\textit{Method}(\mathtt{x}^d \otimes \mathtt{T}_i)$

where $\mathtt{x}^d$ is some term in Φ

3.1.3 Set to Hierarchy

$$(\{\mathtt{C}\}, \Psi) \rightarrow (\{\mathtt{H}\}, \Phi)$$

Pre-conditions:

- $\mathtt{C}$ is a term of type $\mathbb{CLASS}$ in Ψ
- $Hierarchy(\mathtt{C})$ is satisfied by $\mathfrak{M}$

Post-conditions:

- $\mathtt{H}$ is a term of type $\mathbb{HIERARCHY}$ in Φ

3.1.4 Hierarchy Expansion

$$(\{\mathtt{H}\}, \Psi) \rightarrow (\{\mathtt{C}^d, \mathtt{r}\}, \Phi)$$

Such that : if $|\mathcal{I}(\mathtt{H})| > 2$ then $d=1$

if $|\mathcal{I}(\mathtt{H})| = 2$ then $d=0$

Pre-conditions:

- $\mathtt{H}$ is a term of type $\mathbb{HIERARCHY}$ in Ψ

Post-conditions:

- $\mathtt{C}^d$ is a term of type $\mathbb{CLASS}$ in Φ
- $\mathcal{I}(\mathtt{H}) = \{\mathcal{I}(\mathtt{r})\} \cup \mathcal{I}(\mathtt{C}^d)$
- $\textsc{Total}(Inherit, \mathtt{C}^d, \mathtt{r})$ in Φ is satisfied by $\mathfrak{M}$

3.1.5 Partition to Hierarchies

$$(\{\mathtt{C}\}, \Psi) \rightarrow (\{\mathtt{H}_1 \ldots \mathtt{H}_n\}, \Phi)$$

Pre-conditions:

- $\mathtt{C}$ is a term of type $\mathbb{CLASS}$ in Ψ

Post-conditions:

- All terms $\mathtt{h}_i$ $1 \leq i \leq n$ introduced are of type $\mathbb{HIERARCHY}$ in Φ
- $\mathcal{I}(\mathtt{C}) = \mathcal{I}(\mathtt{H}_1) \cup \ldots \cup \mathcal{I}(\mathtt{H}_n)$

3.1.6 To bottom

$$(\{\}, \Psi) \rightarrow (\{\}, \perp_{\mathfrak{M}})$$

3.1.7 Introduction

$$(\{\}, \Psi) \rightarrow (\{\mathtt{t}_1{}^d \ldots \mathtt{t}_n{}^d\}, \Phi)$$

Such that : $0 \leq d \leq 1$

Post-conditions:

- $\mathtt{t}_1{}^d \ldots \mathtt{t}_n{}^d$ are terms of any type in Φ

3.2 Abstraction operators

3.2.1 Aggregation

$$(\{\mathtt{t}_1 \ldots \mathtt{t}_n\}, \Psi) \rightarrow (\{\mathtt{T}\}, \Phi)$$

Pre-conditions:

- Terms $\mathtt{t}_1 \ldots \mathtt{t}_n$ are all of type $\mathbb{CLASS}$ or $\mathbb{SIGNATURE}$ in Ψ

Post-conditions:

- $\mathtt{T}$ is a term of the same type as $\mathtt{t}_1 \ldots \mathtt{t}_n$
- $\mathcal{I}(\mathtt{T}) = \{\mathcal{I}(\mathtt{t}_1) \ldots \mathcal{I}(\mathtt{t}_n)\}$

3.2.2 Union

$$(\{\mathtt{T}_1 \ldots \mathtt{T}_n\}, \Psi) \rightarrow (\mathtt{T}, \Phi)$$

Pre-conditions:

- Terms $\mathtt{T}_1 \ldots \mathtt{T}_n$ are all of type $\mathbb{CLASS}$ or $\mathbb{SIGNATURE}$ in Ψ
- $n \geq 2$

Post-conditions:

- $\mathtt{T}$ is a term of the same type as $\mathtt{T}_1 \ldots \mathtt{T}_n$
- $\mathcal{I}(\mathtt{T}) = \mathcal{I}(\mathtt{T}_1) \cup \ldots \cup \mathcal{I}(\mathtt{T}_n)$
- There exists at least one formula with $\mathtt{T}$ of the flowing forms that is satisfied by $\mathfrak{M}$:
  - $\textit{TOTAL}(\textit{BinaryRelation}, \mathtt{x}^d, \mathtt{T})$
  - $\textit{ISOMORPHIC}(\textit{BinaryRelation}, \mathtt{x}^d, \mathtt{T})$
  - $\textit{TOTAL}(\textit{BinaryRelation}, \mathtt{T}, \mathtt{x}^d)$
  - $\textit{ISOMORPHIC}(\textit{BinaryRelation}, \mathtt{T}, \mathtt{x}^d)$
  - $\textit{ALL}(\textit{BinaryRelation}, \mathtt{T})$
  - $\textit{Method}(\mathtt{x}^d \otimes \mathtt{T})$

where $\mathtt{x}^d$ is some term in Φ

3.2.3 Hierarchy to Set

$$(\{\mathtt{H}\}, \Psi) \rightarrow (\{\mathtt{C}\}, \Phi)$$

Pre-conditions:

- $\mathtt{H}$ is a term of type $\mathbb{HIERARCHY}$ in Ψ

Post-conditions:

- $\mathtt{C}$ is a term of type $\mathbb{CLASS}$ in Φ

3.2.4 Collapse to Hierarchy

$$(\{\mathtt{C}^{d}, \mathtt{r}\}, \Psi) \rightarrow (\mathtt{H}, \Phi)$$

Such that : $0 \leq d \leq 1$

Pre-conditions:

- $\mathtt{C}^{d}$ is a term of type $\mathbb{CLASS}$ in Ψ
- $\textsc{Total}(\mathit{Inherit}, \mathtt{C}^{d}, \mathtt{r})$ in Ψ is satisfied by $\mathfrak{M}$

Post-conditions:

- $\mathtt{H}$ is a term of type $\mathbb{HIERARCHY}$ in Φ
- $\mathcal{I}(\mathtt{H}) = \{\mathcal{I}(\mathtt{r})\} \cup \mathcal{I}(\mathtt{C}^{d})$

3.2.5 Hierarchies Union

$$(\{\mathtt{H}_{1} \ldots \mathtt{H}_{n}\}, \Psi) \rightarrow (\mathtt{C}, \Phi)$$

Pre-conditions:

- All terms $\mathtt{h}i$, $1 \leq i \leq n$ introduced are of type $\mathbb{HIERARCHY}$ in Ψ

Post-conditions:

- $\mathtt{C}$ is a term of type $\mathbb{CLASS}$ in Φ
- $\mathcal{I}(\mathtt{C}) = \mathcal{I}(\mathtt{H}_{1}) \cup \ldots \cup \mathcal{I}(\mathtt{H}_{n})$

3.2.6 To Top

$$(\{\}, \Psi) \rightarrow (\{\}, \top_{\mathfrak{M}})$$

3.2.7 Elimination

$$(\{t_1^d \ldots t_n^d\}, \Psi) \rightarrow (\{\ \}, \Phi)$$

Such that : $0 \leq d \leq 1$

Pre-conditions:

- $t_1^d \ldots t_n^d$ are terms of any type in Ψ

Acknowledgments

We would like to thank Prof. Raymond Turner for his valuable help, guidance and feedback, Jonathan Nicholson for his feedback and nightlong discussions and Christina Maniati for her support.

References

S. Burris and H. P. Sankappanavar. *A course in Universal Algebra*. Springer-Verlag, 1981.

A. H. Eden. *Object-Oriented Modelling*. Under preparation, draft 2007.

A. H. Eden, E. Gasparis, and J. Nicholson. “LePUS3 and Class-Z Reference Manual”. Technical report CSM-474, ISSN 1744-8050, University of Essex, 2007.

M. Manzano. *Model Theory*. Oxford Science Publications, 1999.