A Higher Order Theory of Well-Founded Sets (with Urelements)

Roger Bishop Jones

Abstract

This is a modification of the pure set theory GS to admit urelements.

Created 2010/10/08

Last Change Date: 2013/01/03 17:12:44

http://www.rbjones.com/rbjpub/pp/doc/t042.pdf

Id: t042.doc,v 1.8 2013/01/03 17:12:44 rbj Exp

Contents

1 Introduction

Since this is a modification of a previous formalisation of higher order set theory [?], I omit the preminary discussions for which the reader may refer to the previous document, and confine this introduction to the modifications which I have made in this version.

This is simply the admission of urelements of arbitrary type, so that we introduce here instead of a simple type, a type constructor, which will be called ''a GSU'.

SML

```
open_theory "rbjmisc";
force_new_theory "gsu-ax";
new_parent "U_orders";
new_parent "wf_relp";
new_parent "wf_recp";
force_new_pc "'gsu-ax";
merge_pcs ["'savedthm_cs_∃_proof"] "'gsu-ax";
set_merge_pcs ["basic_hol", "'gsu-ax"];
new_type ("GSU", 1);
```

1.1 Urelements

The novelty is urelements. Since the theory is not conservative over plain HOL, it must be introduced using axioms.

The axiom which introduces the urelements asserts that there is an injection from type ⌜:'a⌝ into type ⌜:'a GSU⌝.

SML

```
val Urelement_Axiom = new_axiom(["Urelement_Axiom"], ⌜∃f:'a → 'a GSU• OneOne f⌝);
```

Having asserted the existence of such an injection we now introduce a constant with that characteristic:

HOL Constant

```
Urelement : 'a → 'a GSU
─────────────────────────
OneOne Urelement
```

The range of this injection is the extension of the set of urelements, but we can't say this until we have introduced membership.

HOL Constant

```
UeVal : 'a GSU → 'a
─────────────────────────
∀x• UeVal x = εy• Urelement y = x
```

```
UeVal_Urelement_lemma = ⊢ ∀ x• UeVal (Urelement x) = x
```

I will use the term Set_u exclusively for bona-fide sets, i.e. values outside the range of this function.

The following predicate is true just of the urelements.

HOL Constant

$\boldsymbol{Ue} : {'a}\ GSU \rightarrow BOOL$

$\forall x\bullet\ Ue\ x \Leftrightarrow \exists y\bullet\ x = Urelement\ y$

$\boldsymbol{Urelement\_Ue\_lemma} = \vdash \forall\ x\bullet\ Ue\ x \Rightarrow Urelement\ (UeVal\ x) = x$

And this one of sets.

HOL Constant

$\boldsymbol{Set_u} : {'a}\ GSU \rightarrow BOOL$

$\forall x\bullet\ Set_u\ x \Leftrightarrow \neg\ \exists y\bullet\ x = Urelement\ y$

$\boldsymbol{UeSet_u\_lemma1} = \vdash \forall x\bullet\ Ue\ x \Leftrightarrow \neg\ Set_u\ x$
$\boldsymbol{Urelement\_Ue\_lemma2} = \vdash \forall\ x\bullet\ \neg\ Set_u\ x \Rightarrow Urelement\ (UeVal\ x) = x$

1.2 Membership

Membership is a relation over the type. We can't define this constant (in this context) so it is introduced as a new constant (about which nothing is asserted except its name and type) and its properties are introduced axiomatically.

SML

```
new_const ("∈u", ⌜:'a GSU → 'a GSU → BOOL⌝);
declare_infix (230,"∈u");
```

Since we have urelements, which are not bona-fide sets, it will be convenient to insist that only sets have members:

SML

```
val Setu_axiom = new_axiom (["Setu_axiom"], ⌜∀x y• x ∈u y ⇒ Setu y⌝);
```

I will possibly be making use of two different treatments of well-foundedness (from the theories *U_orders*, and *wf_relp*) and it may be helpful to establish the connection between them.

The following theorem does the trick:

$\boldsymbol{UWellFounded\_well\_founded\_thm} =$
$\quad \vdash \forall\ \$<<\bullet\ UWellFounded\ \$<< \Leftrightarrow well\_founded\ \$<<$

The axioms of extensionality and well-foundedness may be thought of as telling us what kind of thing a set is (later axioms tell us which sets are to be found in our domain of discourse).

This is a principle point of departure from the theory without urelements. Here I have to chose between preserving extensionality (which can by done using Quine's trick of identifying a urelement with its unit set), or preserving well-foundedness of the membership relation (by insisting that urelements have no members).[1]

[1] I could also fudge it by saying nothing about the membership of urelements, but that seems the least attractive option since both extensionality and well-foundedness would have to be qualified.

When I first addressed this issue, I was mistakenly under the impression that this was just a question of an aribitrary choice of what to say about the membership of urelements, and under this illusion I tried having two different membership relations one well-founded and the other extensional. One could have two different membership relations which differed only what the members of the urelements are, but the adoption of the unqualified axiom of extensionality (which is the point of Quine's trick) is nevertheless substantive, for otherwise, even though a urelement would be its own sole member, it would nevertheless be distinct from its own unit set and extensionality would fail. It therefore seems that an unqualified extensionality is incompatible with the closure of the universe under the formation of unit *sets*, for if Quine's trick is used to admit extensionality, the "unit set" of a urelement will not be a set at all (or else the urelement is also a set).

This consideration persuaded me against urelements being unit sets, and I will therefore have to put up with extensionality being conditional.

1.2.1 Extensionality

The most fundamental property of membership (or is it of sets?) is *extensionality*, which tells us what kind of thing a set is. The axiom tells us that if two sets have the same elements then they are in fact the same set.

SML

```
val gsu_ext_axiom = new_axiom (["gsu_ext_axiom"],
        ⌜∀s t:'a GSU• Set_u s ∧ Set_u t ⇒ (s = t ⇔ ∀e• e ∈_u s ⇔ e ∈_u t)⌝);
```

This may be thought of as extensionality of sets themselves or as extensionality of equality over sets. Though sets are extensional, we do not have an unconditionally extensional equality over the domain of discourse, because we have urelements.

```
gsu_ext_thm =
        ⊢ ∀ s t• Set_u s ⇒ Set_u t ⇒ (s = t ⇔ (∀ e• e ∈_u s ⇔ e ∈_u t))
```

The following (rather crude) conversion is useful in the application of extensionality:

SML

```
fun gsu_ext_conv tm =
        let fun aux thms =
                let val (lhs, rhs) = dest_eq tm;
                    val extax_thm = list_∀_elim [lhs, rhs] gsu_ext_thm;
                    val [ant1, ant2, con] = strip_⇒ (concl extax_thm);
                    val a1thm1 = conv_rule (LEFT_C (TRY_C (rewrite_conv thms))) (disch_rule ant1 (as
                    val a2thm1 = conv_rule (LEFT_C (TRY_C (rewrite_conv thms))) (disch_rule ant2 (as
                    val a1thm2 = try (fn x => ⇒_elim x t_thm) a1thm1;
                    val a2thm2 = try (fn x => ⇒_elim x t_thm) a2thm1;
                    val a1thm3 = try undisch_rule a1thm2;
                    val a2thm3 = try undisch_rule a2thm2;
                    val con_thm1 = ⇒_elim extax_thm a1thm3
                    val con_thm2 = ⇒_elim con_thm1 a2thm3
                in pure_rewrite_conv [con_thm2] tm
                end
        in aux []
        end handle _ => fail_conv tm;
```

The corresponding rule and tactics are:

SML

```
val gsu_ext_rule = conv_rule gsu_ext_conv;
val gsu_ext_tac = conv_tac gsu_ext_conv;
```

Those only work for equations at the top level, the following tactic is provided for equations lower down. It expects to be supplied the term to which it will be applied.

SML

```
val gsu_ext_tac2 = fn tm => rewrite_tac [gsu_ext_conv tm];
```

For facility of reasoning in this theory it is best if as few theorems as possible are conditional upon whether the variables in them have values which are sets rather than urelements. This is achieved firstly by having no members for urelements (which are therefore extensionally equivalent to the empty set). Because of this it is important in introducing new operations which are intended to deliver sets that the result is not specified purely by extension, it is important that the result be known to be a set rather than an urelement even when its extension is empty. We also ensure that all operators over sets are extensional, i.e. the result depends only upon the extension of the arguments, not upon anything else (the only other thing it could depend on would be whether the arguments are sets or urelements). It would be natural for example to make the Subset relation false if either operand was a urelement, but this would lengthen proofs.

It follows from the definitions of *Urelement*, *Ue* and Set_u that nothing is both a set and a urelement, and that urelements are equal iff the values from which they were obtained under *Ue* are equal.

It is convenient to have a function which gives the extension of a $'aGSU$ set as a SET of $'aGSU$s.

HOL Constant

$\boldsymbol{X_u} : 'a\ GSU \to 'a\ GSU\ SET$

$\forall s \bullet\ X_u\ s = \{t \mid t \in_u s\}$

Since equality is not strictly extensional, it is useful to define an extensional equality (equivalence).

SML

```
declare_infix(200, "=u");
```

HOL Constant

$\$ =_{\boldsymbol{u}} : 'a\ GSU \to 'a\ GSU \to BOOL$

$\forall s\ t \bullet\ s =_u t \Leftrightarrow X_u\ s = X_u\ t$

$\boldsymbol{X_u\_thm} = \quad \vdash \forall\ s\ t \bullet\ s \in X_u\ t = s \in_u t$

$\boldsymbol{eq_u\_refl\_thm} = \quad \vdash \forall\ s \bullet\ s =_u s$

$\boldsymbol{eq_u\_sym\_thm} = \quad \vdash \forall\ s\ t \bullet\ s =_u t \Rightarrow t =_u s$

$\boldsymbol{eq_u\_trans\_thm} = \quad \vdash \forall\ s\ t\ u \bullet\ s =_u t \land t =_u u \Rightarrow t =_u u$

$\boldsymbol{eq_u\_ext\_thm} = \vdash \forall\ s\ t \bullet\ s =_u t \Leftrightarrow (\forall\ u \bullet\ u \in_u s \Leftrightarrow u \in_u t)$

$\boldsymbol{\neg eq_u\_\neg eq\_thm} = \quad \vdash \forall\ s\ t \bullet\ \neg\ s =_u t \Rightarrow \neg\ s = t$

1.2.2 Well-Foundedness

Wellfoundedness is asserted using the definition in the theory "U_orders", which is conventional in asserting that each non-empty set has a minimal element.

SML

$val\ \boldsymbol{gsu\_wf\_axiom} = new\_axiom\ (["gsu\_wf\_axiom"], \ulcorner UWellFounded\ \$\in_u \urcorner);$

$\boldsymbol{gsu\_wf\_thm1} = \quad \vdash well\_founded\ \$\in_u$
$\boldsymbol{gsu\_wf\_min\_thm} = \quad \vdash \forall\ x\bullet\ (\exists\ y\bullet\ y \in_u x)$
$\quad \Rightarrow (\exists\ z\bullet\ z \in_u x \land \neg (\exists\ v\bullet\ v \in_u z \land v \in_u x))$
$\boldsymbol{gsu\_wftc\_thm} = \quad \vdash well\_founded\ (tc\ \$\in_u)$

SML

$declare\_infix\ (230, "\in_u{}^+");$

HOL Constant

$\$\in_{\boldsymbol{u}}{}^+ : 'a\ GSU \to 'a\ GSU \to BOOL$

$\$\in_u{}^+ = tc\ \$\in_u$

$\boldsymbol{gsu\_wftc\_thm2} = \quad \vdash well\_founded\ \$\in_u{}^+$
$\boldsymbol{tc\in_u\_incr\_thm} = \quad \vdash \forall\ x\ y\bullet\ x \in_u y \Rightarrow x \in_u{}^+ y$
$\boldsymbol{tc\in_u\_cases\_thm} = \quad \vdash \forall\ x\ y\bullet\ x \in_u{}^+ y \Leftrightarrow (x \in_u y \lor (\exists\ z\bullet\ x \in_u{}^+ z \land z \in_u y))$
$\boldsymbol{tc\in_u\_trans\_thm} = \quad \vdash \forall\ s\ t\ u\bullet\ s \in_u{}^+ t \land t \in_u{}^+ u \Rightarrow s \in_u{}^+ u$
$\boldsymbol{tc\in_u\_decomp\_thm} = \vdash \forall\ x\ y\bullet\ x \in_u{}^+ y \land \neg\ x \in_u y \Rightarrow (\exists\ z\bullet\ x \in_u{}^+ z \land z \in_u y)$
$\boldsymbol{tc\in_u\_decomp\_thm5} = \quad \vdash \forall\ x\ y\bullet\ x \in_u{}^+ y \land \neg\ x \in_u y \Rightarrow (\exists\ z\bullet\ x \in_u z \land z \in_u{}^+ y)$

The following operator restricts a function over sets to a domain determined by some set. It was originally introduced to use in definitions by transfinite recursion, and a recursion principle is later proven for that purpose, but I then had some better ideas on how to define functions.

SML

$declare\_infix\ (300, "\lhd\in^+{}_u");$

HOL Constant

$\$\lhd\in^+{}_{\boldsymbol{u}} : 'a\ GSU \to ('a\ GSU \to 'b) \to ('a\ GSU \to 'b)$

$\forall s\ f\bullet\ s \lhd\in^+{}_u f = \lambda x\bullet\ if\ x \in_u{}^+ s\ then\ f\ x\ else\ \epsilon y\bullet T$

The resulting induction principle (sometimes called Noetherian induction?) is useful.

$\boldsymbol{gsu\_wf\_ind\_thm} = \quad \vdash \forall\ p\bullet\ (\forall\ x\bullet\ (\forall\ y\bullet\ y \in_u x \Rightarrow p\ y) \Rightarrow p\ x) \Rightarrow (\forall\ x\bullet\ p\ x)$
$\boldsymbol{gsu\_cv\_ind\_thm} = \quad \vdash \forall\ p\bullet\ (\forall\ x\bullet\ (\forall\ y\bullet\ tc\ \$\in_u\ y\ x \Rightarrow p\ y) \Rightarrow p\ x) \Rightarrow (\forall\ x\bullet\ p\ x)$
$\boldsymbol{gsu\_cv\_ind\_thm2} = \quad \vdash \forall\ p\bullet\ (\forall\ x\bullet\ (\forall\ y\bullet\ y \in_u{}^+ x \Rightarrow p\ y) \Rightarrow p\ x) \Rightarrow (\forall\ x\bullet\ p\ x)$

But we can get induction tactics directly from the well-foundedness theorems:

SML

```
val 'a GSU_INDUCTION_T = WF_INDUCTION_T gsu_wf_thm1;
val gsu_induction_tac = wf_induction_tac gsu_wf_thm1;
val 'a GSU_INDUCTION_T2 = WF_INDUCTION_T gsu_wftc_thm2;
val gsu_induction_tac2 = wf_induction_tac gsu_wftc_thm2;
```

```
wf_ul1 = ⊢ ∀ x:'a GSU• ¬ x ∈u x
wf_ul2 = ⊢ ∀ x y:'a GSU• ¬ (x ∈u y ∧ y ∈u x)
wf_ul3 = ⊢ ∀ x y z:'a GSU• ¬ (x ∈u y ∧ y ∈u z ∧ z ∈u x)
```

1.3 The Ontology Axiom

The remaining axioms are intended to ensure that the sets are a large and well-rounded part of the cumulative heirarchy. This is to be accomplished by defining a Galaxy as a set satisfying certain closure properties and then asserting that every set is a member of some Galaxy. It is convenient to introduce new constants and axioms for each of the Galactic closure properties before asserting the existence of the Galaxies.

Here we define the Subset relation and assert the existence of powersets and unions.

1.3.1 Subsets

A Subset s of t is a set all of whose members are also members of t.

SML

```
declare_infix (230,"⊆u");
declare_infix (230,"⊂u");
```

HOL Constant

```
$⊆u : 'a GSU → 'a GSU → BOOL
──────────────────────────────
∀s t• s ⊆u t ⇔ ∀e• e ∈u s ⇒ e ∈u t
```

HOL Constant

```
$⊂u : 'a GSU → 'a GSU → BOOL
──────────────────────────────
∀s t• s ⊂u t ⇔ s ⊆u t ∧ ¬ t ⊆u s
```

```
⊆u_eq_thm =      ⊢ ∀ A B• Setu A ∧ Setu B ⇒ (A = B ⇔ A ⊆u B ∧ B ⊆u A)
⊆u_refl_thm =    ⊢ ∀ A• A ⊆u A
∈u⊆u_def =       ⊢ ∀ e A B• e ∈u A ∧ A ⊆u B ⇒ e ∈u B
⊆u_trans_thm =   ⊢ ∀ A B C• A ⊆u B ∧ B ⊆u C ⇒ A ⊆u C
⊂u_trans_thm =   ⊢ ∀ A B C• A ⊂u B ∧ B ⊂u C ⇒ A ⊂u C
not_⊂u_thm = ⊢ ∀ x• ¬ x ⊂u x
```

HOL Constant

```
⊆u_closed : 'a GSU → BOOL
──────────────────────────────
∀s• ⊆u_closed s ⇔ ∀e f• e ∈u s ∧ f ⊆u e ⇒ f ∈u s
```

1.3.2 The Ontology Axiom

We now specify with a single axiom the closure requirements which ensure that our universe is adequately populated. The ontology axiom states that every set is a member of some galaxy which is transitive and closed under formation of powersets and unions and under replacement.

The formulation of replacement only makes membership of a galaxy dependent on the range being contained in the galaxy, it asserts unconditionally the sethood of the image of a set under a functional relation.

Because we have urelements and the ontology axiom introduces sets by their extension, special provision is necessary to ensure the existence of the empty set. In the corresponding theory without urelements the existence of the empty set is obtained from the ontology axiom using the clause which corresponds to the axiom of replacement, but this axiom only establishes that something has no members, leaving open the possibility that there is an urelement but there is no empty set. We therefore assert that the image of a something through a functional relation is a set.

In some other places it is necessary to insist on certain objects being sets. Thus, power sets never contain urelements, and this must be made explicit. This would not have been necessary had we defined the subset relation as holding only between sets, which might possibly have been better.

It is not necessary to assert sethood in any other case, however, I have found it expedient to mention sethood in a couple of places where it is strictly redundant. This is because to get the consistency proof for the specifications of the constants used for these constructors I would otherwise need to know that the empty set exists, so that I can insist on them yielding it whenever the result has an empty extension. However, I can't prove the existence of the empty set until I have the separation axiom. Well I could, but...

SML

```
val UOntology_axiom =
        new_axiom (["UOntology_axiom"],
⌜ ∀s:'a GSU•
        ∃g• s ∈u g
∧
        ∀t• t ∈u g
        ⇒ t ⊆u g
        ∧ (∃p• (∀v• v ∈u p ⇔ Setu v ∧ v ⊆u t) ∧ p ∈u g ∧ Setu p)
        ∧ (∃u• (∀v• v ∈u u ⇔ ∃w• v ∈u w ∧ w ∈u t) ∧ u ∈u g ∧ Setu u)
        ∧ (∀rl• ManyOne rl ⇒
                (∃r• (∀v• v ∈u r ⇔ ∃w • w ∈u t ∧ rl w v)
                        ∧ (r ⊆u g ⇒ r ∈u g)
                        ∧ Setu r))⌝
);
```

I originally thought that in this version of set theory with urelements, iteration of the type constructor would always create larger universes. However, I don't see anything which tells us that there is a set containing all the urelements, and it is therefore possible that the urelements have the same cardinality as the sets.

1.4 PowerSets and Union

Here we define the powerset and union operators.

1.4.1 PowerSets

HOL Constant

$\mathbb{P}_{\boldsymbol{u}}$: $'a\ GSU \to 'a\ GSU$

$\forall s\bullet\ Set_u\ (\mathbb{P}_u\ s) \wedge \forall t{:}'a\ GSU\bullet\ t \in_u \mathbb{P}_u\ s \Leftrightarrow Set_u\ t \wedge t \subseteq_u s$

$\mathbb{P}_{\boldsymbol{u}}$**_thm** $= \vdash \forall\ s\ t\bullet\ t \in_u \mathbb{P}_u\ s = (Set_u\ t \wedge t \subseteq_u s)$
s$\in\mathbb{P}_{\boldsymbol{us}}$**_thm** $= \ \vdash \forall\ s\bullet\ Set_u\ s \Rightarrow s \in_u \mathbb{P}_u\ s$
stc$\in\mathbb{P}_{\boldsymbol{us}}$**_thm** $= \vdash \forall\ s\bullet\ Set_u\ s \Rightarrow s \in_u{}^+\ \mathbb{P}_u\ s$
Set$_{\boldsymbol{u}}\mathbb{P}_{\boldsymbol{u}}$**_thm** $= \vdash \forall\ s\bullet\ Set_u\ (\mathbb{P}_u\ s)$
eq$\mathbb{P}_{\boldsymbol{u}}$**_thm** $=$
$\vdash \forall\ s\ t\bullet\ Set_u\ s \wedge Set_u\ t$
$\Rightarrow ((s = \mathbb{P}_u\ t) \Leftrightarrow (\forall\ x\bullet\ x \in_u s \Leftrightarrow Set_u\ x \wedge x \subseteq_u t))$

1.4.2 Union

HOL Constant

$\bigcup_{\boldsymbol{u}}$: $'a\ GSU \to 'a\ GSU$

$\forall s\bullet\ Set_u\ (\bigcup_u\ s) \wedge \forall t\bullet\ t \in_u \bigcup_u\ s \Leftrightarrow \exists e\bullet\ t \in_u e \wedge e \in_u s$

$\bigcup_{\boldsymbol{u}}$**_thm** $=\vdash \forall\ s\ t\bullet\ t \in_u \bigcup_u\ s \Leftrightarrow (\exists\ e\bullet\ t \in_u e \wedge e \in_u s)$
tc$\in_{\boldsymbol{u}}$_$\bigcup_{\boldsymbol{u}}$**_thm** $=\vdash \forall\ s\ t\bullet\ t \in_u{}^+\ \bigcup_u\ s \Leftrightarrow (\exists\ e\bullet\ t \in_u{}^+\ e \wedge e \in_u s)$
$\in_{\boldsymbol{u}}\bigcup_{\boldsymbol{u}}$**_thm** $= \ \vdash \forall s\ t{:}'a\ GSU\bullet\ Set_u\ t \wedge t \in_u s \Rightarrow t \subseteq_u \bigcup_u\ s$
$\in_{\boldsymbol{u}}\bigcup_{\boldsymbol{u}}$**_thm2** $= \ \vdash \forall\ s\ t\bullet\ t \in_u \bigcup_u\ s \Rightarrow (\exists\ e\bullet\ t \in_u e \wedge e \in_u s)$
$\in_{\boldsymbol{u}}\bigcup_{\boldsymbol{u}}$**_thm3** $= \ \vdash \forall\ s\ t\bullet\ (\exists\ e\bullet\ t \in_u e \wedge e \in_u s) \Rightarrow t \in_u \bigcup_u\ s$
$\bigcup_{\boldsymbol{u}}$**_ext_thm** $= \vdash \forall\ x\ y\bullet\ (\bigcup_u\ x = y) \Leftrightarrow (Set_u\ y \wedge (\forall\ z\bullet\ z \in_u y \Leftrightarrow (\exists\ w\bullet\ z \in_u w \wedge w \in_u x)))$
Set$_{\boldsymbol{u}}\bigcup_{\boldsymbol{u}}$**_thm** $= \vdash \forall\ s\bullet\ Set_u\ (\bigcup_u\ s)$

1.5 Relational Replacement

The constant $RelIm_u$ is defined for relational replacement.

HOL Constant

RelIm$_{\boldsymbol{u}}$: $('a\ GSU \to 'a\ GSU \to BOOL) \to 'a\ GSU \to 'a\ GSU$

$\forall rl\ s\bullet\ Set_u\ (RelIm_u\ rl\ s) \wedge (ManyOne\ rl \Rightarrow (\forall t\bullet\ t \in_u RelIm_u\ rl\ s \Leftrightarrow \exists e\bullet\ e \in_u s \wedge rl\ e\ t))$

Set$_{\boldsymbol{uRelImu}}$**_thm** $= \vdash \forall\ rel\ s\bullet\ Set_u\ (RelIm_u\ rel\ s)$
RelIm$_{\boldsymbol{u}}$**_thm** $=$
$\vdash \forall\ rl\ s\bullet\ ManyOne\ rl \Rightarrow (\forall\ t\bullet\ t \in_u RelIm_u\ rl\ s = (\exists\ e\bullet\ e \in_u s \wedge rl\ e\ t))$

1.6 Separation

Separation is introduced by conservative extension.

The specification of Sep_u which follows is introduced after proving that it is satisfied by a term involving the use of $RelIm_u$.

This higher order formulation of separation is accomplished by defining a new constant which takes a property of sets p and a set s and returns the Subset of s consisting of those elements which satisfy p.

HOL Constant

$\boldsymbol{Sep_u} : 'a\ GSU \to ('a\ GSU \to BOOL) \to 'a\ GSU$

$\forall s\ p\bullet\ (\forall e\bullet e \in_u (Sep_u\ s\ p) \Leftrightarrow e \in_u s \land p\ e) \land Set_u\ (Sep_u\ s\ p)$

$\boldsymbol{Sep_u\_thm} = \vdash \forall s\ p\ e\bullet e \in_u (Sep_u\ s\ p) \Leftrightarrow e \in_u s \land p\ e$
$\boldsymbol{Set_u\_Sep_u\_thm} = \vdash \forall s\ p\bullet\ Set_u\ (Sep_u\ s\ p)$

$\boldsymbol{Sep_u\_\subseteq_u\_thm} = \vdash \forall\ s\ p\bullet\ Set_u\ s \Rightarrow Sep_u\ s\ p \subseteq_u s$
$\boldsymbol{Sep_u\_sub\_thm} = \vdash \forall\ s\ p\ e\bullet\ e \in_u Sep_u\ s\ p \Rightarrow e \in_u s$
$\boldsymbol{Sep_u\_\in_u\_\mathbb{P}_u\_thm} = \vdash \forall\ s\ p\bullet\ Set_u\ s \Rightarrow Sep_u\ s\ p \in_u \mathbb{P}_u\ s$
$\boldsymbol{Sep_u\_\subseteq\_thm} = \vdash \forall\ s\ t\bullet\ t \subseteq_u s \Rightarrow Sep_u\ s\ (CombC\ \$\in_u\ t) = t$

1.7 Galaxies

A Galaxy is a transitive set closed under powerset formation, union and replacement. The Ontology axioms ensures that every set is a member of some galaxy. Here we define a galaxy constructor and establish some of its properties.

1.7.1 Definition of Galaxy

First we define the property of being a galaxy.

HOL Constant

$\boldsymbol{galaxy_u}: 'a\ GSU \to BOOL$

$$\begin{aligned}
&\forall s\bullet\\
&\quad galaxy_u\ s \Leftrightarrow (\exists x\bullet\ x \in_u s)\\
&\quad \land \forall t\bullet\ t \in_u s\\
&\qquad \Rightarrow t \subseteq_u s\\
&\qquad \land \mathbb{P}_u\ t \in_u s\\
&\qquad \land \bigcup_u t \in_u s\\
&\qquad \land (\forall rl\bullet\ ManyOne\ rl\\
&\qquad\quad \Rightarrow RelIm_u\ rl\ t \subseteq_u s\\
&\qquad\quad \Rightarrow RelIm_u\ rl\ t \in_u s)
\end{aligned}$$

$\boldsymbol{galaxies_u\_\exists\_thm} =$
$\quad \vdash \forall s\bullet\ \exists g\bullet\ s \in_u g \land galaxy_u\ g$

1.7.2 Definition of Gx

Gx_u is a function which maps each set onto the smallest galaxy of which it is a member.

HOL Constant

$\boldsymbol{Gx_u}$: $'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ t\bullet\ t \in_u Gx_u\ s \Leftrightarrow \forall g\bullet\ galaxy_u\ g \wedge s \in_u g \Rightarrow t \in_u g$

Each set is in its smallest enclosing galaxy, which is of course a galaxy and is contained in any other galaxy of which that set is a member:

$\boldsymbol{t\_in\_Gx_u\_t\_thm}$ = $\vdash \forall\ t\bullet\ t \in_u Gx_u\ t$
$\boldsymbol{tc\in_u\_Gx_u\_thm}$ = $\vdash \forall\ t\bullet\ t \in_u{}^+ Gx_u\ t$
$\boldsymbol{Set_u\_Gx_u\_thm}$ = $\vdash \forall\ x\bullet\ Set_u\ (Gx_u\ x)$

$\boldsymbol{Gx_u\_\subseteq_u\_galaxy_u}$ = $\vdash \forall s\ g\bullet\ galaxy_u\ g \wedge s \in_u g \Rightarrow (Gx_u\ s) \subseteq_u g$
$\boldsymbol{galaxy_u\_Gx_u}$ = $\vdash \forall s\bullet\ galaxy_u\ (Gx_u\ s)$

1.7.3 Galaxy Closure

The galaxy axiom asserts that a Galaxy is a transitive set closed under construction of powersets, distributed union and replacement. Galaxies are also closed under most other ways of constructing sets, and we need to demonstrate these facts systematically as the theory is developed.

HOL Constant

$\boldsymbol{Transitive_u}$: $'a\ GSU \rightarrow BOOL$

$\forall s\bullet\ Transitive_u\ s \Leftrightarrow \forall e\bullet\ e \in_u s \Rightarrow e \subseteq_u s$

$\boldsymbol{galaxies_u\_transitive\_thm}$ = $\vdash \forall s\bullet\ galaxy_u\ s \Rightarrow Transitive_u\ s$

$\boldsymbol{G_{uTrans}\_thm}$ = $\vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall\ s\ t\bullet\ s \in_u g \wedge t \in_u s \Rightarrow t \in_u g)$
$\boldsymbol{GClose_u\mathbb{P}_u\_thm}$ = $\vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall\ s\bullet\ s \in_u g \Rightarrow \mathbb{P}_u\ s \in_u g)$
$\boldsymbol{GClose_u\bigcup\_thm}$ = $\vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall\ s\bullet\ s \in_u g \Rightarrow \bigcup_u\ s \in_u g)$
$\boldsymbol{GClose_{uSepu}\_thm}$ = $\vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall s\bullet\ s \in_u g \Rightarrow \forall p\bullet\ Sep_u\ s\ p \in_u g)$
$\boldsymbol{GClose_u\_\subseteq\_thm}$ = $\vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall\ s\bullet\ s \in_u g \Rightarrow (\forall\ t\bullet\ Set_u\ t \wedge t \subseteq_u s \Rightarrow t \in_u g))$

$\boldsymbol{GClose_u\_fc\_clauses}$ =
$\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\Rightarrow (\forall\ s$
$\bullet\ s \in_u g$
$\Rightarrow \mathbb{P}_u\ s \in_u g$
$\wedge \bigcup_u\ s \in_u g$
$\wedge (\forall\ p\bullet\ Sep_u\ s\ p \in_u g)$
$\wedge (\forall\ t\bullet\ Set_u\ t \wedge t \subseteq_u s \Rightarrow t \in_u g))$

$tc\in_u\_lemma =$ $\vdash \forall\ s\ e\bullet\ e \in_u{}^+\ s \Rightarrow (\forall\ t\bullet\ Transitive_u\ t \land s \subseteq_u t \Rightarrow e \in_u t)$
$GClose_u\_tc\in_u\_thm =$ $\vdash \forall\ s\ g\bullet\ galaxy_u\ g \Rightarrow s \in_u{}^+\ g \Rightarrow s \in_u g$
$GClose_u\_tc\in_u\_thm2 =$ $\vdash \forall\ t\ s\ g\bullet\ galaxy_u\ g \land t \in_u g \land s \in_u{}^+\ t \Rightarrow s \in_u g$

$Gx_u\_mono\_thm =$ $\vdash \forall s\ t\bullet\ s \subseteq_u t \Rightarrow Gx_u\ s \subseteq_u Gx_u\ t$
$Gx_u\_mono\_thm2 =$ $\vdash \forall s\ t\bullet\ s \in_u t \Rightarrow Gx_u\ s \subseteq_u Gx_u\ t$
$t\_\subseteq_u\_Gx_u\_t\_thm =$ $\vdash \forall\ t\bullet\ t \subseteq_u Gx_u\ t$
$Gx_u\_mono\_thm3 =$ $\vdash \forall\ s\ t\bullet\ s \subseteq_u t \Rightarrow s \subseteq_u Gx_u\ t$
$Gx_u\_mono\_thm4 =$ $\vdash \forall\ s\ t\bullet\ Set_u\ s \land s \subseteq_u t \Rightarrow s \in_u Gx_u\ t$

$Gx_u\_trans\_thm =$ $\vdash \forall\ s\bullet\ Transitive_u\ (Gx_u\ s)$
$Gx_u\_trans\_thm2 =$ $\vdash \forall\ s\ t\bullet\ s \in_u t \Rightarrow s \in_u Gx_u\ t$
$Gx_u\_trans\_thm3 =$ $\vdash \forall\ s\ t\ u\bullet\ s \in_u t \land t \in_u Gx_u\ u \Rightarrow s \in_u Gx_u\ u$
$Gx_u\_trans\_thm4 =$ $\vdash \forall\ s\ t\ u\bullet\ s \in_u{}^+\ t \land t \in_u Gx_u\ u \Rightarrow s \in_u Gx_u\ u$
$Gx_u\_trans\_thm5 =$ $\vdash \forall\ s\ t\ u\bullet\ s \in_u{}^+\ t \Rightarrow s \in_u Gx_u\ t$

1.7.4 The Empty Set

We can now prove that there is an empty set.

So we define $\ulcorner \varnothing_u \urcorner$ as the empty set:

HOL Constant

$\varnothing_u : {}'a\ GSU$

$Set_u\ \varnothing_u \land \forall s\bullet\ \neg\ s \in_u \varnothing_u$

and the empty set is a member of every galaxy:

$\varnothing_u\_thm =$ $\vdash \forall\ s\bullet\ \neg\ s \in_u \varnothing_u$
$tc\in_u\_\varnothing_u\_thm =$ $\vdash \forall\ x\bullet\ \neg\ x \in_u{}^+\ \varnothing_u$
$X\varnothing_u\_thm =$ $\vdash \forall\ x\bullet\ \neg\ x \in X_u\ \varnothing_u$
$eq\_\varnothing_u\_\neg\in_u\_thm =$ $\vdash \forall\ x\bullet\ x =_u \varnothing_u \Rightarrow (\forall\ y\bullet\ \neg\ y \in_u x)$
$eq\_\varnothing_u\_\neg tc\in_u\_thm =$ $\vdash \forall\ x\bullet\ x =_u \varnothing_u \Rightarrow (\forall\ y\bullet\ \neg\ y \in_u{}^+\ x)$
$eq_u\_eq\_\varnothing_u\_thm =$ $\vdash \forall\ \alpha\bullet\ Set_u\ \alpha \land \alpha =_u \varnothing_u \Rightarrow \alpha = \varnothing_u$

$Set_u\_\varnothing_u\_thm =$ $\vdash Set_u\ \varnothing_u$
$G\varnothing_{uC} =$ $\vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow \varnothing_u \in_u g$
$\varnothing_u\subseteq_u\_thm =$ $\vdash \forall\ s\bullet\ \varnothing_u \subseteq_u s$
$\bigcup_u\varnothing_u\_thm =$ $\vdash \bigcup_u\ \varnothing_u = \varnothing_u$

$\in_u\_not\_empty\_thm =$ $\vdash \forall\ m\ n\bullet\ m \in_u n \Rightarrow \neg\ n = \varnothing_u$
$\varnothing_u\_\in_u\_galaxy_u\_thm =$ $\vdash \forall\ x\bullet\ galaxy_u\ x \Rightarrow \varnothing_u \in_u x$
$\varnothing_u\_\in_u\_Gx_u\_thm =$ $\vdash \forall\ x\bullet\ \varnothing_u \in_u Gx_u\ x$

It is sometimes useful to force a value to a set having the same extension. The following function is the identity over sets and maps all urelements to the empty set.

HOL Constant

$\boldsymbol{set_u} : 'a\ GSU \to 'a\ GSU$

$\forall x\bullet\ set_u\ x = if\ Set_u\ x\ then\ x\ else\ \varnothing_u$

$\boldsymbol{Set_u\_set_u\_thm} =$
$\quad \vdash \forall\ s\bullet\ Set_u\ (set_u\ s)$
$\boldsymbol{set_u\_eq_u\_thm} =$
$\quad \vdash \forall\ s\bullet\ set_u\ s =_u s$
$\boldsymbol{set_u\_eq_u\_thm2} =$
$\quad \vdash \forall\ s\ u\bullet\ u \in_u set_u\ s \Leftrightarrow u \in_u s$
$\boldsymbol{set_u\_fc\_thm} =$
$\quad \vdash \forall\ s\bullet\ Set_u\ s \Rightarrow set_u\ s = s$

1.8 Functional Replacement

The more convenient form of replacement which takes a function rather than a relation (and hence needs no "ManyOne" caveat) is introduced here.

1.8.1 Introduction

Though a functional formulation of replacement is most convenient for most applications, it has a number of small disadvantages which have persuaded me to stay closer to the traditional formulation of replacement in terms of relations. The more convenient functional version will then be introduced by definition (so if you don't know what I'm talking about, look forward to compare the two versions).

For the record the disadvantages of the functional one (if used as an axiom) are:

1. It can't be used to prove the existence of the empty set.
2. When used to define separation it requires the axiom of choice.

Now we prove a more convenient version of replacement which takes a HOL function rather than a relation as its argument. It states that the image of any set under a function is also a set.

$\ulcorner Imagep_u\ f\ s\urcorner$ is the image of s through f.

HOL Constant

$\boldsymbol{Imagep_u} : ('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$

$\forall f\ s\bullet\ Set_u(Imagep_u\ f\ s) \land \forall x\bullet\ x \in_u Imagep_u\ f\ s \Leftrightarrow \exists e\bullet\ e \in_u s \land x = f\ e$

This is what computer scientists might call Map.

$\boldsymbol{Imagep_u\_\emptyset_u\_thm}$ =
$\vdash \forall\ f \bullet\ Imagep_u\ f\ \emptyset_u\ =\ \emptyset_u$
$\boldsymbol{tc\in_u\_Imagep_u\_thm}$ =
$\vdash \forall\ f\ s\ x \bullet\ x \in_u{}^+\ Imagep_u\ f\ s \Leftrightarrow (\exists\ y \bullet\ y \in_u\ s\ \wedge\ (x\ =\ f\ y\ \vee\ x \in_u{}^+\ f\ y))$

Replacement is used later for definitions by transfinite induction.

$\boldsymbol{Imagep_u\_comp\_thm}$ =
$\vdash \forall\ s\ f\ g \bullet\ Imagep_u\ f\ (Imagep_u\ g\ s)\ =\ Imagep_u\ (f\ o\ g)\ s$

1.8.2 Galaxy Closure

We now show that galaxies are closed under $Imagep_u$.

$\boldsymbol{GImagep_{uC}}\ =\ \vdash \forall g \bullet\ galaxy_u\ g \Rightarrow \forall s \bullet\ s \in_u\ g$
$\Rightarrow \forall f \bullet\ Imagep_u\ f\ s \subseteq_u\ g \Rightarrow Imagep_u\ f\ s \in_u\ g$

1.9 Pair and Unit sets

$Pair_u$ is defined using replacement, and $Unit_u$ using $Pair_u$.

$Pair_u$ can be defined as the image of some two element set under a function defined by a conditional. A suitable two element set can be constructed from the empty set using the powerset construction a couple of times. However, having proven that this can be done (details omitted), we can introduce the pair constructor by conservative extension as follows. (the ProofPower tool shows that it has accepted my proof by putting this extension into the "definitions" section of the theory listing).

HOL Constant

$\boldsymbol{Pair_u}$: $'a\ GSU \rightarrow\ 'a\ GSU \rightarrow\ 'a\ GSU$

$\forall s\ t{:}'a\ GSU \bullet\ Set_u\ (Pair_u\ s\ t)\ \wedge\ \forall e{:}'a\ GSU \bullet\ e \in_u\ Pair_u\ s\ t \Leftrightarrow e\ =\ s\ \vee\ e\ =\ t$

$\boldsymbol{Pair_u\_\in_u\_thm}$ = $\vdash \forall\ x\ y \bullet\ x \in_u\ Pair_u\ x\ y\ \wedge\ y \in_u\ Pair_u\ x\ y$
$\boldsymbol{Pair_u\_tc\in_u\_thm}$ = $\vdash \forall\ s\ t \bullet\ s \in_u{}^+\ Pair_u\ s\ t\ \wedge\ t \in_u{}^+\ Pair_u\ s\ t$
$\boldsymbol{Pair_u\_eq\_thm}$ = $\vdash \forall\ s\ t\ u\ v \bullet\ Pair_u\ s\ t\ =\ Pair_u\ u\ v$
$\Leftrightarrow\ s\ =\ u\ \wedge\ t\ =\ v\ \vee\ s\ =\ v\ \wedge\ t\ =\ u$

$\boldsymbol{GClose_{uPairu}}$ = $\vdash \forall g \bullet\ galaxy_u\ g \Rightarrow \forall s\ t \bullet\ s \in_u\ g\ \wedge\ t \in_u\ g$
$\Rightarrow Pair_u\ s\ t \in_u\ g$

HOL Constant

$\boldsymbol{Unit_u}$: $'a\ GSU \rightarrow\ 'a\ GSU$

$\forall s \bullet\ Unit_u\ s\ =\ Pair_u\ s\ s$

The following theorem tells you what the members of a unit sets are.

$|$ $\boldsymbol{Unit_u\_thm}$ = $\vdash \forall s\ e\bullet\ e \in_u Unit_u\ s \Leftrightarrow e = s$
$|$ $\boldsymbol{Unit_u\_thm2}$ = $\vdash \forall x\bullet\ x \in_u Unit_u\ x$
$|$ $\boldsymbol{Unit_u\_tc\in_u\_thm}$ = $\vdash \forall x\bullet\ x \in_u{}^+\ Unit_u\ x$

The following theorem tells you when two unit sets are equal.

$|$ $\boldsymbol{Unit_u\_eq\_thm}$ = $\vdash \forall s\ t\bullet\ Unit_u\ s = Unit_u\ t \Leftrightarrow s = t$

1.9.1 Galaxy Closure

$|$ $\boldsymbol{GClose_{uUnitu}}$ = $\vdash \forall g\bullet\ galaxy_u\ g \Rightarrow \forall s\bullet\ s \in_u g \Rightarrow Unit_u\ s \in_u g$

The following theorems tell you when $Pair_u$ are really $Unit_s$.

$|$ $\boldsymbol{Unit_u\_Pair_u\_eq\_thm}$ = $\vdash \forall s\ t\ u\bullet\ Unit_u\ s = Pair_u\ t\ u \Leftrightarrow s = t \land s = u$
$|$ $\boldsymbol{Pair_u\_Unit_u\_eq\_thm}$ = $\vdash \forall s\ t\ u\bullet\ Pair_u\ s\ t = Unit_u\ u \Leftrightarrow s = u \land t = u$

1.10 Union and Intersection

Binary union and distributed and binary intersection are defined.

1.10.1 Binary Union

HOL Constant

$\$\cup_u : 'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ t\bullet\ Set_u\ (s \cup_u t) \land \forall e\bullet\ e \in_u (s \cup_u t) \Leftrightarrow e \in_u s \lor e \in_u t$

$|$ $\boldsymbol{\subseteq_u\cup_u\_thm}$ = $\vdash \forall\ A\ B\bullet\ A \subseteq_u A \cup_u B \land B \subseteq_u A \cup_u B$
$|$ $\boldsymbol{\cup_u\subseteq_u\_def1}$ = $\vdash \forall\ A\ B\ C\bullet\ A \subseteq_u C \land B \subseteq_u C \Rightarrow A \cup_u B \subseteq_u C$
$|$ $\boldsymbol{\cup_u\subseteq_u\_def2}$ = $\vdash \forall\ A\ B\ C\ D\bullet\ A \subseteq_u C \land B \subseteq_u D \Rightarrow A \cup_u B \subseteq_u C \cup_u D$
$|$ $\boldsymbol{\cup_u\varnothing_u\_clauses}$ = $\vdash \forall\ A\bullet\ Set_u\ A \Rightarrow A \cup_u \varnothing_u = A \land \varnothing_u \cup_u A = A$
$|$ $\boldsymbol{\cup_u\_comm\_thm}$ = $\vdash \forall\ A\ B\bullet\ A \cup_u B = B \cup_u A$
$|$ $\boldsymbol{\cup_u\_\varnothing_u\_set_u\_thm}$ = $\vdash \forall\ A\bullet\ A \cup_u \varnothing_u = set_u\ A \land \varnothing_u \cup_u A = set_u\ A$
$|$ $\boldsymbol{tc\in_u\_\cup_u\_thm}$ = $\vdash \forall\ x\ A\ B\bullet\ x \in_u{}^+\ A \cup_u B \Leftrightarrow x \in_u{}^+\ A \lor x \in_u{}^+\ B$

1.10.2 Galaxy Closure

$|$ $\boldsymbol{GClose_u\cup_u}$ = $\vdash \forall g\bullet\ galaxy_u\ g \Rightarrow \forall s\ t\bullet\ s \in_u g \land t \in_u g \Rightarrow s \cup_u t \in_u g$

1.10.3 Distributed Intersection

Distributed intersection doesn't really make sense for the empty set, but under this definition it maps the empty set onto itself.

HOL Constant

$\bigcap_{\boldsymbol{u}} : 'a\ GSU \to 'a\ GSU$

$\forall s \bullet\ \bigcap_u\ s\ =\ Sep_u\ (\bigcup_u\ s)\ (\lambda x \bullet\ \forall t \bullet\ t \in_u s \Rightarrow x \in_u t)$

$\boldsymbol{Set_u \bigcap_u\_thm} = \vdash \forall\ x \bullet\ Set_u\ (\bigcap_u\ x)$
$\boldsymbol{\bigcap_u \subseteq_u\_thm} = \vdash \forall x\ s\ e \bullet\ x \in_u s$
$\quad \Rightarrow (e \in_u \bigcap_u\ s \Leftrightarrow \forall y \bullet\ y \in_u s \Rightarrow e \in_u y)$
$\boldsymbol{\subseteq_u \bigcap_u\_thm} = \vdash \forall A\ B \bullet\ A \in_u B$
$\quad \Rightarrow \forall C \bullet\ (\forall D \bullet\ D \in_u B \Rightarrow C \subseteq_u D)$
$\quad \Rightarrow C \subseteq_u \bigcap_u\ B$
$\boldsymbol{\bigcap_u \varnothing_u\_thm} = \vdash \bigcap_u\ \varnothing_u = \varnothing_u$

1.10.4 Binary Intersection

Binary intersection could be defined in terms of distributed intersection, but its easier not to.

SML

```
declare_infix (240, "∩u");
```

HOL Constant

$\$\cap_{\boldsymbol{u}} : 'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall s\ t \bullet\ s \cap_u t = Sep_u\ s\ (\lambda x \bullet\ x \in_u t)$

$\boldsymbol{Set_u \cap_u\_thm} = \vdash \forall\ s\ t \bullet\ Set_u\ (s \cap_u t)$

1.10.5 Galaxy Closure

$\boldsymbol{GClose_u \bigcap_u} = \vdash \forall g \bullet\ galaxy_u\ g \Rightarrow \forall s \bullet\ s \in_u g \Rightarrow \bigcap_u\ s \in_u g$
$\boldsymbol{GClose_u \cap_u} = \vdash \forall g \bullet\ galaxy_u\ g \Rightarrow \forall s\ t \bullet\ s \in_u g \wedge t \in_u g \Rightarrow s \cap_u t \in_u g$

$\boldsymbol{\cap_u\_thm} = \vdash \forall s\ t\ e \bullet\ e \in_u s \cap_u t \Leftrightarrow e \in_u s \wedge e \in_u t$
$\boldsymbol{\cap_u\_thm} = \vdash \forall s\ t\ e \bullet\ e \in_u s \cap_u t \Leftrightarrow e \in_u s \wedge e \in_u t$

$\boldsymbol{\subseteq_u \cap_u\_thm} = \vdash \forall A\ B \bullet\ A \cap_u B \subseteq_u A \wedge A \cap_u B \subseteq_u B$
$\boldsymbol{\cap_u \subseteq_u\_def1} = \vdash \forall A\ B\ C \bullet\ A \subseteq_u C \wedge B \subseteq_u C \Rightarrow A \cap_u B \subseteq_u C$
$\boldsymbol{\cap_u \subseteq_u\_def2} = \vdash \forall A\ B\ C\ D \bullet\ A \subseteq_u C \wedge B \subseteq_u D \Rightarrow (A \cap_u B) \subseteq_u (C \cap_u D)$
$\boldsymbol{\cap_u \subseteq_u\_def3} = \vdash \forall A\ B\ C \bullet\ C \subseteq_u A \wedge C \subseteq_u B \Rightarrow C \subseteq_u A \cap_u B$

1.10.6 Consequences of Well-Foundedness

$\boldsymbol{not\_x\_\in_u\_x\_thm} = \vdash \neg\ (\exists\ x \bullet\ x \in_u x)$

1.11 Galaxy Closure Clauses

$\boldsymbol{GClose_u\_fc\_clauses2}$ =

$\vdash \forall\ g$

$\bullet\ galaxy_u\ g$

$\Rightarrow (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow Pair_u\ s\ t \in_u g)$

$\land (\forall\ s \bullet\ s \in_u g \Rightarrow Unit_u\ s \in_u g)$

$\land (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow s \cup_u t \in_u g)$

$\land (\forall\ s \bullet\ s \in_u g \Rightarrow \bigcap_u s \in_u g)$

$\land (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow s \cap_u t \in_u g)$

$\boldsymbol{tc\in_u\_clauses}$ = $\vdash \forall\ s \bullet\ s \in_u{}^+ Unit_u\ s$

$\land \forall t \bullet\ t \in_u{}^+ Pair_u\ s\ t$

$\land\ s \in_u{}^+ Pair_u\ s\ t$

1.12 Transitive Closure and Closed Image

The transitive closure of a set is the set of objects which relate to it under the transitive closure of the membership relation.

HOL Constant

$\boldsymbol{TrCl_u} : 'a\ GSU \rightarrow 'a\ GSU$

$\forall s \bullet\ TrCl_u\ s = Sep_u\ (Gx_u\ s)\ (\lambda x \bullet\ x \in_u{}^+ s)$

$\boldsymbol{Set_u\_TrCl_u\_thm}$ =

$\vdash \forall\ s \bullet\ Set_u\ (TrCl_u\ s)$

$\boldsymbol{TrCl_u\_sup\_thm}$ =

$\vdash \forall\ s \bullet\ s \subseteq_u TrCl_u\ s$

$\boldsymbol{TrCl_u\_sup\_thm2}$ =

$\vdash \forall\ s\ t \bullet\ Transitive_u\ t \land s \subseteq_u t \Rightarrow TrCl_u\ s \subseteq_u t$

$\boldsymbol{Transitive_u\_TrCl_u\_thm}$ =

$\vdash \forall\ s \bullet\ Transitive_u\ (TrCl_u\ s)$

$\boldsymbol{TrCl_u\_ext\_thm}$ =

$\vdash \forall\ s\ x \bullet\ x \in_u TrCl_u\ s \Leftrightarrow (\forall\ t \bullet\ Transitive_u\ t \land s \subseteq_u t \Rightarrow x \in_u t)$

$\boldsymbol{TrCl_u\_ext\_thm2}$ =

$\vdash \forall\ s\ t \bullet\ s \in_u TrCl_u\ t \Leftrightarrow s \in_u{}^+ t$

$\boldsymbol{tc\in_u\_TrCl_u\_thm}$ =

$\vdash \forall\ s\ t \bullet\ s \in_u{}^+ TrCl_u\ t \Leftrightarrow s \in_u{}^+ t$

$\boldsymbol{Tran\_set\_TrCl\_thm}$ =

$\vdash \forall\ s \bullet\ Set_u\ s \land Transitive_u\ s \Rightarrow TrCl_u\ s = s$

$\boldsymbol{Tran\_set\_tc\in_u\_thm}$ =

$\vdash \forall\ s \bullet\ Set_u\ s \land Transitive_u\ s \Rightarrow (\forall\ x \bullet\ x \in_u{}^+ s \Rightarrow x \in_u s)$

$\boldsymbol{Tran\_tc\in_u\_thm}$ =

$\vdash \forall\ s \bullet\ Transitive_u\ s \Rightarrow (\forall\ x \bullet\ x \in_u{}^+ s \Rightarrow x \in_u s)$

1.12.1 Closed Image

When I first introduced the following operator I expected it to be more widely applicable than turns out to be the case. It is now likely to be phased out in favour of the following limit construction.

Transitive closure is useful later in combination with functional replacement for defining operators over ordinals. To facilitate such definitions a *closed image* function is defined, which delivers the transitive closure of a set obtained by functional replacement.

Think of $ClIm_u$ as the (transitively) CLosed IMage.

HOL Constant

$\boldsymbol{ClIm_u} : ('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$

$\forall f\ \alpha\bullet\ ClIm_u\ f\ \alpha\ =\ TrCl_u(Imagep_u\ f\ \alpha)$

The first thing we need to "prove" is that it yields a set, and then its membership conditions.

$\boldsymbol{Set_u\_ClIm_u\_thm}$ =
$\vdash \forall\ f\ \alpha\bullet\ Set_u\ (ClIm_u\ f\ \alpha)$
$\boldsymbol{\in_u\_ClIm_u\_thm}$ =
$\vdash \forall\ f\ \alpha\bullet\ x \in_u ClIm_u\ f\ \alpha \Leftrightarrow (\exists\ y\bullet\ y \in_u \alpha \wedge (x = f\ y \vee x \in_u{}^+ f\ y))$
$\boldsymbol{tc\in_u\_ClIm_u\_thm}$ =
$\vdash \forall\ f\ \alpha\ x\bullet\ x \in_u{}^+ ClIm_u\ f\ \alpha \Leftrightarrow (\exists\ y\bullet\ y \in_u \alpha \wedge (x = f\ y \vee x \in_u{}^+ f\ y))$

This will be used for definitions by transfinite induction, in which case the "base" case will correspond to images of the empty set, so these are not interesting. When empty images are excluded there is an analogy for composition of these maps with ordinary functional composition, which results in an associative law. If this is proven at this level it subsequently yields more specific associative laws, e.g. for ordinal addition and multiplication.

$\boldsymbol{ClIm_u\_\varnothing_u\_thm}$ =
$\vdash \forall\ f\ s\bullet\ Set_u\ s \Rightarrow (ClIm_u\ f\ s = \varnothing_u \Leftrightarrow s = \varnothing_u)$
$\boldsymbol{ClIm_u\_\varnothing_u\_thm2}$ =
$\vdash \forall\ f\ s\bullet\ ClIm_u\ f\ s = \varnothing_u \Leftrightarrow s =_u \varnothing_u$
$\boldsymbol{ClIm_u\_\varnothing_u\_thm3}$ =
$\vdash \forall\ f\bullet\ ClIm_u\ f\ \varnothing_u = \varnothing_u$
$\boldsymbol{ClIm_u\_ext\_thm}$ =
$\vdash \forall\ f\ s\ t\bullet\ s =_u t \Rightarrow ClIm_u\ f\ s = ClIm_u\ f\ t$
$\boldsymbol{ClIm_u\_mono\_thm}$ =
$\vdash \forall\ f\ s\ t\bullet\ s \subseteq_u t \Rightarrow ClIm_u\ f\ s \subseteq_u ClIm_u\ f\ t$

Whereas the associative behaviour extends to arbitrary functions under functional replacement, in this special kind of replacement it extends only to functions which are well-behaved in relation to transitive closure. We therefore define the notion of "closure compatibility" for which property we use the name *ClCo*, as follows.

HOL Constant

$\boldsymbol{ClCo_u} : ('a\ GSU \to 'a\ GSU) \to BOOL$

$\forall f\bullet\ ClCo_u\ f \Leftrightarrow \forall x\ y\bullet\ x \in_u{}^+ y \Rightarrow f\ x \in_u{}^+ f\ y$

$\boldsymbol{ClCo_u\_ClIm_u\_thm}$ =

$\quad \vdash \forall\ s\ f\ g\bullet\ ClCo_u\ f \Rightarrow ClIm_u\ f\ (ClIm_u\ g\ s) = ClIm_u\ (f\ o\ g)\ s$

1.12.2 Limits

In defining operations over ordinals it will be useful to take the limit of a family of sets. When we come to the ordinals that will be a family of ordinals, but the limit operation can be defined more generally simply as the union of the family. This pre-ordinal development is placed here since it is another way of applying the replacement principle.

HOL Constant

$\boldsymbol{Limit_u} : ('a\ GSU \rightarrow 'a\ GSU) \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall f\ \alpha\bullet\ Limit_u\ f\ \alpha = \bigcup_u(Imagep_u\ f\ \alpha)$

The first thing we need to "prove" is that it yields a set, and then its membership conditions.

$\boldsymbol{Set_u\_Limit_u\_thm}$ =

$\quad \vdash \forall\ f\ \alpha\bullet\ Set_u\ (Limit_u\ f\ \alpha)$

$\boldsymbol{\in_u\_Limit_u\_thm}$ =

$\quad \vdash \forall\ f\ \alpha\bullet\ x \in_u Limit_u\ f\ \alpha \Leftrightarrow (\exists\ y\bullet\ y \in_u \alpha \wedge x \in_u f\ y)$

$\boldsymbol{tc\in_u\_Limit_u\_thm}$ =

$\quad \vdash \forall\ f\ \alpha\ x\bullet\ x \in_u{}^+ Limit_u\ f\ \alpha \Leftrightarrow (\exists\ y\bullet\ y \in_u \alpha \wedge x \in_u{}^+ f\ y)$

1.13 Recursion Theorems

To facilitate definition by transfinite induction induction theorems are required which permit various kinds of inductive definition to be proven consistent automatically.

Certain practices in the presentation of inductive definitions facilitate such proofs. In general, rather than referring directly in the body of such a definition to the function being defined, it is desirable to refer to restricted version of the function obtained by restricting the function to some domain related to the argument of the function by a relationship which is well-founded.

The simplest case of this is to restrict the function to the transitive closure of the set it is applied to. The two other cases are the use of functional replacement, i.e. the image of a set under a function, and the closed image, i.e. the transitive closure of the image of a set under a function.

To get a recursion theorem we prove first that the functional used in the theorem respects the ordering by membership.

$\boldsymbol{\lhd tc\in_u\_respects\_\in_u\_thm}$ =

$\quad \vdash \forall\ af\bullet\ (\lambda\ f\ x\bullet\ af\ (x \lhd\in^+{}_u\ f)\ x)\ respects\ \$\in_u$

$\boldsymbol{Imagep_u\_respects\_\in_u\_thm}$ =

$\quad \vdash \forall\ af\bullet\ (\lambda\ f\ x\bullet\ af\ (Imagep_u\ f\ x)\ x)\ respects\ \$\in_u$

$\boldsymbol{ClIm_u\_respects\_\in_u\_thm}$ =

$\quad \vdash \forall\ af\bullet\ (\lambda\ f\ x\bullet\ af\ (ClIm_u\ f\ x)\ x)\ respects\ \$\in_u$

The automatic existence proof facility is geared to working with functions over recursive datatypes with constructor functions, and though we don't really want a constructor for this application, it won't work without one. So we use *CombI*, which is an identity function, as a dummy constructor to trigger the consistency proof.

$\lhd tc\in_{u}\_recursion\_thm$ =
$\vdash \forall\ af\bullet\ \exists\ f\bullet\ \forall\ x\bullet\ f\ (CombI\ x) = af\ (x \lhd\in^{+}{}_{u}\ f)\ x$

$Imagep_{u}\_recursion\_thm$ =
$\vdash \forall\ af\bullet\ \exists\ f\bullet\ \forall\ x\bullet\ f\ (CombI\ x) = af\ (Imagep_{u}\ f\ x)\ x$

$ClIm_{u}\_recursion\_thm$ =
$\vdash \forall\ af\bullet\ \exists\ f\bullet\ \forall\ x\bullet\ f\ (CombI\ x) = af\ (ClIm_{u}\ f\ x)\ x$

Now we plug in the recursion theorems for use in automatic consistency proofs. Each is put in a distinct proof context, that particular context is used for consistency proofs of definitions using that recursion principle. Because of the hack using *CombI* they cannot all be put in the same proof context.

SML
```
force_new_pc "'gsu−rec1";
force_new_pc "'gsu−rec2";
force_new_pc "'gsu−rec3";
add_∃_cd_thms [Imagepu_recursion_thm] "'gsu−rec1";
add_∃_cd_thms [◁tc∈u_recursion_thm] "'gsu−rec2";
add_∃_cd_thms [ClImu_recursion_thm] "'gsu−rec3";
```

1.14 Set Clauses

For the application of the principle of extensionality we need to know that the expressions on either side of the equality are sets.

The following theorem is useful in automating the proofs.

1.15 Proof Context

To simplify Subsequent proofs a new "proof context" is created enabling automatic use of the results now available.

1.15.1 Principles

The only principle I know of which assists with elementary proofs in set theory is the principle that set theoretic conjectures can be reduced to the predicate calculus by using extensional rules for relations and for operators.

Too hasty a reduction can be overkill and may convert a simple conjecture into an unintelligible morass. We have sometimes in the past used made available two proof contexts, an aggressive extensional one, and a milder non-extensional one. However, the extensional rules for the operators are fairly harmless, expansion is triggered by the extensional rules for the relations (equality and Subset), so a proof context containing the former together with a suitable theorem for the latter gives good control.

1.15.2 Theorems Used Recklessly

This is pretty much guesswork, only time will tell whether this is the best collection.

SML

```
val gsu_ax_thms = [
        ∅u_thm,
        get_spec ⌜ℙu⌝,
        ⋃u_def,
        Imagepu_thm,
        Imagepu_∅u_thm,
        tc∈u_Imagepu_thm,
        Pairu_eq_thm,
        Pairu_def,
        Unitu_eq_thm,
        Unitu_thm,
        Pairu_Unitu_eq_thm,
        Unitu_Pairu_eq_thm,
        Sepu_thm,
        ∪u_def,
        ∩u_thm
];

val gsu_opext_clauses =
        (all_∀_intro
        o list_∧_intro
        o (map all_∀_elim))
        gsu_ax_thms;
save_thm ("gsu_opext_clauses", gsu_opext_clauses);
```

1.15.3 Theorems Used Cautiously

The following theorems are too aggressive for general use in the proof context but are needed when attempting automatic proof. When an extensional proof is appropriate it can be initiated by a cautious (i.e. a "once") rewrite using the following clauses, after which the extensional rules in the proof context will be triggered.

[This used to be just two extensionality theorems, but one is no longer so its a unit list.]

SML

```
val gsu_relext_clauses =
        (all_∀_intro
        o list_∧_intro
        o (map all_∀_elim))
        [⊆u_def, equ_ext_thm];
save_thm ("gsu_relext_clauses", gsu_relext_clauses);
```

There are a number of important theorems, such as well-foundedness and galaxy closure which have not been mentioned in this context. The character of these theorems makes them unsuitable for the proof context, their application requires thought.

1.15.4 Automatic Proof

The basic proof automation is augmented by adding a preliminary rewrite with the relational extensionality clauses.

SML

```
fun gsu_ax_prove_conv thl =
        TRY_C (pure_rewrite_conv [gsu_relext_clauses])
        THEN_TRY_C gsu_ext_conv
        THEN_C (basic_prove_conv thl);
```

1.15.5 Proof Context 'gsu-ax

SML

```
val nostu_thms = [
        galaxyu_Gxu,
        t_in_Gxu_t_thm,
        SetuImagepu_thm,
        Imagepu_∅u_thm,
        Setuℙu_thm,
        Setu⋃u_thm,
        SetuRelImu_thm,
        Setu_Sepu_thm,
        SetuUnitu_thm,
        Setu⋂u_thm,
        Setu∩u_thm,
        setu_equ_thm,
        setu_equ_thm2,
        Setu_TrClu_thm,
        ∈u_ClImu_thm,
        tc∈u_ClImu_thm,
        TrClu_ext_thm2,
        tc∈u_TrClu_thm,
        tc∈u_∪u_thm,
```

```
        tc∈u-⋃u-thm
];

add_rw_thms (gsu_ax_thms @ nost_u_thms) "'gsu−ax";
add_sc_thms (gsu_ax_thms @ nost_u_thms) "'gsu−ax";
add_st_thms gsu_ax_thms "'gsu−ax";
set_pr_conv gsu_ax_prove_conv "'gsu−ax";
set_pr_tac
        (conv_tac o gsu_ax_prove_conv)
        "'gsu−ax";
commit_pc "'gsu−ax";
```

Using the proof context "'gsu-ax" elementary results in gsu are now provable automatically on demand. For this reason it is not necessary to prove in advance of needing them results such as the associativity of intersection, since they can be proven when required by an expression of the form "prove rule[] term" which proves *term* and returns it as a theorem. If the required proof context for doing this is not in place the form " *merge_pcs_rule* ["*basic_hol*", '*gsu − ax*"] (*prove_rule* []) *term*" may be used. Since this is a little cumbersome we define the function *gsu_ax_rule* and illustrate its use as follows:

SML

```
val gsu_ax_rule =
        (merge_pcs_rule1
        ["basic_hol", "'gsu−ax"]
        prove_rule) [];
val gsu_ax_conv =
        MERGE_PCS_C1
        ["basic_hol", "'gsu−ax"]
        prove_conv;
val gsu_ax_tac =
        conv_tac o gsu_ax_conv;
```

1.15.6 Examples

The following are examples of the elementary results which are now proven automatically:

SML

```
gsu_ax_rule ⌜
        a ∩u (b ∩u c)
        = (a ∩u b) ∩u c⌝;
gsu_ax_rule ⌜
        a ∩u (b ∩u c)
        = (a ∩u b) ∩u c⌝;
gsu_ax_rule ⌜a ∩u b ⊆u b⌝;
(∗ gsu_ax_rule ⌜∅u ∪u b = b⌝; ∗)
gsu_ax_rule ⌜
        a ⊆u b ∧ c ⊆u d
```

```
|       ⇒ a ∩_u c ⊆_u b ∩_u d⌝;
|gsu_ax_rule ⌜Sep_u b p ⊆_u b⌝;
|gsu_ax_rule ⌜a ⊆_u b ⇒
|       Imagep_u f a ⊆_u Imagep_u f b⌝;
```

2 Products and Sums

A new "gsu-fun" theory is created as a child of "gsu-ax". The theory will contain the definitions of ordered pairs, cartesian product, relations and functions, dependent products (functions), dependent sums (disjoint unions) and related material for general use.

SML

```
open_theory "gsu−ax";
force_new_theory "gsu−fun";
force_new_pc "′gsu−fun";
merge_pcs ["′savedthm_cs_∃_proof"] "′gsu−fun";
set_merge_pcs ["basic_hol", "′gsu−ax", "′gsu−fun"];
```

2.1 Ordered Pairs

SML

```
declare_infix (240,"↦_u");
```

I first attempted to define ordered pairs in a more abstract way than by explicit use of the Wiener-Kuratovski representation, but this gace me problems so I eventually switched to the explicit definition.

This influences the development of the theory, since the first thing I do is to replicate the previously used defining properties.

HOL Constant

```
 $↦_u : 'a GSU → 'a GSU → 'a GSU
─────────────────────────────────
 ∀s t• (s ↦_u t) = Pair_u (Unit_u s) (Pair_u s t)
```

```
Set_u↦_u_thm =                ⊢ ∀ s t• Set_u (s ↦_u t)
↦_u_eq_thm =                  ⊢ ∀ s t u v• (s ↦_u t = u ↦_u v) = (s = u ∧ t = v)
Pair_u_∈_u_↦_u_thm =          ⊢ ∀s t• Pair_u s t ∈_u s ↦_u t
Pair_u_∈_u_Gx_u_↦_u_thm =     ⊢ ∀ s t• Pair_u s t ∈_u Gx_u (s ↦_u t)
↦_u_spec_thm =                ⊢ (∀ s t u v• (s ↦_u t = u ↦_u v) = (s = u ∧ t = v))
                              ∧ (∀ s t• Pair_u s t ∈_u s ↦_u t)
                              ∧ (∀ s t• Pair_u s t ∈_u Gx_u (s ↦_u t))
↦_u_∈_u_Gx_u_Pair_u_thm =     ⊢ ∀ s t• s ↦_u t ∈_u Gx_u (Pair_u s t)
```

```
¬↦_u∅_u_thm = ⊢ ∀ x y• ¬ x ↦_u y = ∅_u
¬∅_u↦_u_thm = ⊢ ∀ x y• ¬ ∅_u = x ↦_u y
GClose_u↦_u_thm = ⊢ ∀ g• galaxy_u g ⇒ (∀ s t• s ∈_u g ∧ t ∈_u g ⇒ s ↦_u t ∈_u g)
```

$tc\in_u\_\mapsto_u\_left\_thm = \vdash \forall\ s\ t\bullet\ s \in_u{}^+\ s \mapsto_u t$
$tc\in_u\_\mapsto_u\_right\_thm = \vdash \forall\ s\ t\bullet\ t \in_u{}^+\ s \mapsto_u t$

2.1.1 MkPair and MkTriple

It proves convenient to have constructors which take HOL pairs and triples as parameters.

HOL Constant

$MkPair_u : 'a\ GSU \times 'a\ GSU \to 'a\ GSU$

$\forall lr\bullet\ MkPair_u\ lr = (Fst\ lr) \mapsto_u (Snd\ lr)$

HOL Constant

$MkTriple_u : 'a\ GSU \times 'a\ GSU \times 'a\ GSU \to 'a\ GSU$

$\forall t\bullet\ MkTriple_u\ t = (Fst\ t) \mapsto_u (MkPair_u\ (Snd\ t))$

2.2 Relations

HOL Constant

$Rel_u : 'a\ GSU \to BOOL$

$\forall x\bullet\ Rel_u\ x \Leftrightarrow \forall y\bullet\ y \in_u x \Rightarrow \exists s\ t\bullet\ y = s \mapsto_u t$

$Rel_u\_\varnothing_u\_thm = \vdash Rel_u\ \varnothing_u$

The domain is the set of elements which are related to something under the relationship.

HOL Constant

$Dom_u : 'a\ GSU \to 'a\ GSU$

$\forall x\bullet\ Dom_u\ x = Sep_u\ (Gx_u\ x)\ (\lambda w\bullet\ \exists v\bullet\ w \mapsto_u v \in_u x)$

$Set_{uDom_u}\_thm = \vdash \forall\ r\bullet\ Set_u\ (Dom_u\ r)$
$Dom_u\_\varnothing_u\_thm = \vdash Dom_u\ \varnothing_u = \varnothing_u$
$Dom_u\_thm = \vdash \forall\ r\ y\bullet\ y \in_u Dom_u\ r \Leftrightarrow (\exists\ x\bullet\ y \mapsto_u x \in_u r)$
$Dom_u\_Gx_u\_thm = \vdash \forall\ r\bullet\ Dom_u\ r \in_u Gx_u\ r$
$GClose_u\_Dom_u\_thm = \vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall\ r\bullet\ r \in_u g \Rightarrow Dom_u\ r \in_u g)$

$tc\in_u\_Dom_u\_thm = \vdash \forall\ x\bullet\ x \in_u{}^+\ Dom_u\ y \Rightarrow x \in_u{}^+\ y$

HOL Constant

$Ran_u : 'a\ GSU \to 'a\ GSU$

$\forall x\bullet\ Ran_u\ x = Sep_u\ (Gx_u\ x)\ (\lambda w\bullet\ \exists v\bullet\ v \mapsto_u w \in_u x)$

$\boldsymbol{Set_{uRanu}\_thm} = \quad \vdash \forall\ s\bullet\ Set_u\ (Ran_u\ s)$
$\boldsymbol{Ran_u\_\varnothing_u\_thm} = \quad \vdash Ran_u\ \varnothing_u\ =\ \varnothing_u$
$\boldsymbol{Ran_u\_thm} = \quad \vdash \forall r\ y\bullet\ y \in_u Ran_u\ r \Leftrightarrow \exists\ x\bullet\ x \mapsto_u y \in_u r$
$\boldsymbol{GClose_u\_Ran_u\_thm} = \quad \vdash \forall\ g\bullet\ galaxy_u\ g \Rightarrow (\forall\ r\bullet\ r \in_u g \Rightarrow Ran_u\ r \in_u g)$
$\boldsymbol{tc\in_u\_Ran_u\_thm} = \quad \vdash \forall\ x\ y\bullet\ x \in_u{}^+\ Ran_u\ y \Rightarrow x \in_u{}^+\ y$

The following function maps a HOL function over the range of a GSU function, returning in effect the composition of the two functions. Its primary use is likely to be with sequences, for which it can be used to systematically transform all the elements of the sequence, but its definitiom is not specific to sequences so it is included here.

HOL Constant

$\boldsymbol{RanMap_u}$: $('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$

$\forall f\ s\bullet\ RanMap_u\ f\ s = Imagep_u\ (\lambda e\bullet\ Fst_u\ e \mapsto_u f\ (Snd_u\ e))\ s$

RanMap does preserves the domain of a relation.

$\boldsymbol{Dom_u\_RanMap_u\_thm} = \vdash \forall\ f\ r\bullet\ Rel_u\ r \Rightarrow Dom_u\ (RanMap_u\ f\ r) = Dom_u\ r$

HOL Constant

$\boldsymbol{Field_u}$: $'a\ GSU \to 'a\ GSU$

$\forall s\bullet\ Field_u\ s = (Dom_u\ s) \cup_u (Ran_u\ s)$

$\boldsymbol{Field_u\_\varnothing_u\_thm} = \quad \vdash Field_u\ \varnothing_u\ =\ \varnothing_u$

$\boldsymbol{tc\in_u\_Field_u\_thm} = \quad \vdash \forall\ x\bullet\ x \in_u{}^+\ Field_u\ y \Rightarrow x \in_u{}^+\ y$

2.3 Domain and Range Restrictions

SML

```
declare_infix (300, "◁u");
declare_infix (300, "▷u");
declare_infix (300, "⩤u");
declare_infix (300, "⩥u");
```

HOL Constant

$\$\lhd_u$: $'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall s\ r\bullet\ s \lhd_u r = Sep_u\ r\ (\lambda p\bullet\ Fst_u\ p \in_u s)$

HOL Constant

$\$\rhd_u$: $'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ r\bullet\ r \rhd_u s = Sep_u\ r\ (\lambda p\bullet\ Snd_u\ p \in_u s)$

HOL Constant

$\$\unlhd_u$: $'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ r\bullet\ s \unlhd_u r = Sep_u\ r\ (\lambda p\bullet\ \neg\ Fst_u\ p \in_u s)$

HOL Constant

$\$\unrhd_u$: $'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ r\bullet\ r \unrhd_u s = Sep_u\ r\ (\lambda p\bullet\ \neg\ Snd_u\ p \in_u s)$

2.4 Dependent Types

Any relation may be regarded as a dependent sum type. When so regarded, each ordered pair in the relation consist with a type-index and a value whose type is that associated with the type.

The indexed set of types, relative to which every pair in the relation is well-typed may be retrieved from the relation as follows.

HOL Constant

$\boldsymbol{Rel2DepType_u} : 'a\ GSU \rightarrow 'a\ GSU$

$$
\begin{array}{l}
\forall r\bullet\ Rel2DepType_u\ r = Sep_u \\
\quad (Gx_u\ r) \\
\quad (\lambda e\bullet\ \exists i\ t{:}'a\ GSU\bullet \\
\qquad e = i \mapsto_u t \\
\qquad \wedge\ i \in_u Dom_u\ r \\
\qquad \wedge\ (\forall j\bullet\ j \in_u t \Leftrightarrow i \mapsto_u j \in_u r))
\end{array}
$$

$\boldsymbol{Set_{uRel2DepTypeu}\text{-}thm} = \vdash \forall\ r\bullet\ Set_u\ (Rel2DepType_u\ r)$

Any similar indexed collection of sets, determines a set of ordered pairs and a set of functions according to the following definitions.

The dependent sums are as follows:

HOL Constant

$\boldsymbol{DepSum_u} : 'a\ GSU \rightarrow 'a\ GSU$

$$
\begin{array}{l}
\forall t\bullet\ DepSum_u\ t = Sep_u \\
\quad (Gx_u\ t) \\
\quad (\lambda e\bullet\ \exists i\ t2\ v{:}'a\ GSU\bullet \\
\qquad e = i \mapsto_u v \\
\qquad \wedge\ v \in_u t2 \\
\qquad \wedge\ i \mapsto_u t2 \in_u t)
\end{array}
$$

$$Set_{uDepSumu}\_thm = \vdash \forall\ t\bullet\ DepSum_u\ t$$
$$= Sep_u$$
$$(Gx_u\ t)$$
$$(\lambda\ e\bullet\ \exists\ i\ t2\ v\bullet\ e = i \mapsto_u v \land v \in_u t2 \land i \mapsto_u t2 \in_u t)$$

2.5 Dependent Sums and Cartesian Products

SML

```
declare_binder "Σu";
```

HOL Constant

$$\$\Sigma_u : ('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$$

$$\forall f\ s\bullet\ \$\Sigma_u\ f\ s = \bigcup_u ($$
$$Imagep_u\quad (\lambda e\bullet\ Imagep_u\ (\lambda x\bullet\ e \mapsto_u x)\ (f\ e))$$
$$s$$
$$)$$

SML

```
declare_infix(240,"×u");
```

HOL Constant

$$\$\times_u : 'a\ GSU \to 'a\ GSU \to 'a\ GSU$$

$$\forall s\ t\bullet\ s \times_u t = \bigcup_u ($$
$$Imagep_u$$
$$(\lambda se\bullet\ (Imagep_u\ (\lambda te\bullet\ se \mapsto_u te)\ t))$$
$$s)$$

$$Set_u\times_u\_thm = \vdash \forall\ s\ t\bullet\ Set_u\ (s \times_u t)$$

$$f{\mapsto}_{us}\_thm =$$
$$\vdash \forall\ s\ t\ p\bullet\ p \in_u s \times_u t \Rightarrow Fst_u\ p \mapsto_u Snd_u\ p = p$$

$$v{\in}_u{\times}_u\_thm =$$
$$\vdash \forall\ p\ s\ t\bullet\ p \in_u s \times_u t \Rightarrow Fst_u\ p \in_u s \land Snd_u\ p \in_u t$$

$${\mapsto}_u{\in}_u{\times}_u\_thm =$$
$$\vdash \forall\ l\ r\ s\ t\bullet\ l \mapsto_u r \in_u s \times_u t \Leftrightarrow (l \in_u s \land r \in_u t)$$

2.5.1 Relation Space

This is the set of all relations over some domain and codomain, i.e. the power set of the cartesian product.

SML

```
declare_infix(240,"↔_u");
```

HOL Constant

```
$↔_u : 'a GSU → 'a GSU → 'a GSU
──────────────────────────────────
∀s t• s ↔_u t = ℙ_u(s ×_u t)
```

```
↔_u⊆_u×_u_thm =      ⊢ ∀s t r• r ∈_u s ↔_u t ⇔ Set_u r ∧ r ⊆_u (s ×_u t)
∅_u∈_u↔_u_thm =      ⊢ ∀s t• ∅_u ∈_u s ↔_u t
```

2.5.2 Another Pair-Projection Inverse Theorem

Couched in terms of membership of relation spaces.

SML

```
set_goal ([], ⌜∀p r s t•
        p ∈_u r ∧
        r ∈_u s ↔_u t ⇒
        Fst_u(p) ↦_u Snd_u(p) = p⌝);
a (prove_tac[
        get_spec ⌜$↔_u⌝,
        ⊆_u_def]);
a (REPEAT
        (asm_fc_tac[f↦_us_thm]));
val f↦_us_thm1 =
        save_pop_thm "f↦_us_thm1";
```

2.5.3 Member of Relation Theorem

SML

```
set_goal ([],⌜∀p r s t•
        p ∈_u r ∧
        r ∈_u s ↔_u t ⇒
        Fst_u(p) ∈_u s ∧
        Snd_u(p) ∈_u t⌝);
a (prove_tac[
        get_spec ⌜$↔_u⌝,
        ⊆_u_def]);
a (asm_fc_tac[]);
a (fc_tac[v∈_u×_u_thm]);
a (asm_fc_tac[]);
a (fc_tac[v∈_u×_u_thm]);
val ∈_u↔_u_thm =
        save_pop_thm "∈_u↔_u_thm";
```

2.5.4 Relational Composition

SML

```
declare_infix (250,"o_u");
```

HOL Constant

$\$o_u : 'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall f\ g \bullet\ f\ o_u\ g =$
$\quad Imagep_u$
$\quad (\lambda p \bullet\ (Fst_u(Fst_u\ p) \mapsto_u Snd_u(Snd_u\ p)))$
$\quad (Sep_u\ (g \times_u f)\ \lambda p \bullet\ \exists q\ r\ s \bullet\ p = (q \mapsto_u r) \mapsto_u (r \mapsto_u s))$

$o_u\_thm =$
$\quad \vdash \forall f\ g\ x \bullet\ x \in_u f\ o_u\ g \Leftrightarrow$
$\quad\quad \exists q\ r\ s \bullet\ q \mapsto_u r \in_u g \wedge r \mapsto_u s \in_u f$
$\quad\quad\quad \wedge\ x = q \mapsto_u s$
$o_u\_thm2 =$
$\quad \vdash \forall\ f\ g\ x\ y \bullet\ x \mapsto_u y \in_u f\ o_u\ g$
$\quad\quad \Leftrightarrow (\exists\ z \bullet\ x \mapsto_u z \in_u g \wedge z \mapsto_u y \in_u f)$

$o_u\_associative\_thm =$
$\quad \vdash \forall f\ g\ h \bullet\ (f\ o_u\ g)\ o_u\ h = f\ o_u\ g\ o_u\ h$

$o_u\_Rel_u\_thm =$
$\quad \vdash \forall\ r\ s \bullet\ Rel_u\ r \wedge Rel_u\ s \Rightarrow Rel_u\ (r\ o_u\ s)$

$\boldsymbol{Set_{uou}\_thm} = \vdash \forall\ r\ s \bullet\ Set_u\ (r\ o_u\ s)$

2.5.5 Relation Subset of Cartesian Product

$Rel_u\_\subseteq_u\_cp\_thm =$
$\quad \vdash \forall\ x \bullet\ Rel_u\ x \Leftrightarrow (\exists\ s\ t \bullet\ x \subseteq_u s \times_u t)$

2.6 Functions

Definition of partial and total functions and the corresponding function spaces.

2.6.1 fun

HOL Constant

$Fun_u : 'a\ GSU \to BOOL$

$\forall x \bullet\ Fun_u\ x \Leftrightarrow Rel_u\ x\ \wedge$
$\quad \forall s\ t\ u \bullet\ s \mapsto_u u \in_u x$
$\quad\quad \wedge\ s \mapsto_u t \in_u x$
$\quad\quad \Rightarrow u = t$

2.6.2 lemmas

$\boldsymbol{Fun_u\_\varnothing_u\_thm}$ =
  $\vdash Fun_u\ \varnothing_u$
$\boldsymbol{o_u\_Fun_u\_thm}$ =
  $\vdash \forall\ f\ g\bullet\ Fun_u\ f\ \wedge\ Fun_u\ g \Rightarrow Fun_u\ (f\ o_u\ g)$
$\boldsymbol{Ran_u\_o_u\_thm}$ =
  $\vdash \forall\ f\ g\bullet\ Ran_u\ (f\ o_u\ g) \subseteq_u Ran_u\ f$
$\boldsymbol{Dom_u\_o_u\_thm}$ =
  $\vdash \forall\ f\ g\bullet\ Dom_u\ (f\ o_u\ g) \subseteq_u Dom_u\ g$
$\boldsymbol{Dom_u\_o_u\_thm2}$ =
  $\vdash \forall\ f\ g\bullet\ Ran_u\ g \subseteq_u Dom_u\ f \Rightarrow Dom_u\ (f\ o_u\ g) = Dom_u\ g$
$\boldsymbol{Fun_u\_RanMap_u\_thm}$ =
  $\vdash \forall\ f\ g\bullet\ Fun_u\ g \Rightarrow Fun_u\ (RanMap_u\ f\ g)$

2.6.3 Partial Function Space

This is the set of all partial functions (i.e. many one mappings) over some domain and codomain.

SML

```
declare_infix (240, "⇸_u");
```

HOL Constant

$\$\nrightarrow_u : 'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ t\bullet\ s \nrightarrow_u t = Sep_u\ (s \leftrightarrow_u t)\ Fun_u$

2.6.4 Partial Function Space Non-Empty

First the theorem that the empty set is a partial function over any domain and codomain.

$\boldsymbol{\varnothing_u\in_u\nrightarrow_u\_thm} = \vdash \forall s\ t\bullet\ \varnothing_u \in_u s \nrightarrow_u t$
$\boldsymbol{\exists\nrightarrow_u\_thm} = \vdash \forall s\ t\bullet\ \exists\ f\bullet\ f \in_u s \nrightarrow_u t$

2.6.5 Function Space

This is the set of all total functions over some domain and codomain.

SML

```
declare_infix (240, "→_u");
```

HOL Constant

$\$\rightarrow_u : 'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\ t\bullet\ s \rightarrow_u t = Sep_u\ (s \nrightarrow_u t)$
  $\lambda r\bullet\ s \subseteq_u (Dom_u\ r)$

2.6.6 Function Space Non-Empty

First, for the special case of function spaces with empty domain we prove the theorem that the empty set is a member: Then that whenever the codomain is non-empty the function space is non-empty.

$\boldsymbol{\varnothing_u \in_u \varnothing_u \rightarrow_u\_thm}$ =
$\vdash \forall s\ t \bullet\ \varnothing_u \in_u \varnothing_u \rightarrow_u t$

$\boldsymbol{\exists \rightarrow_u\_thm}$ =
$\vdash \forall\ s\ t \bullet\ (\exists\ x \bullet\ x \in_u t) \Rightarrow (\exists\ f \bullet\ f \in_u s \rightarrow_u t)$

HOL Constant

$\rightarrow_u\_closed : {}'a\ GSU \rightarrow BOOL$

$\forall s \bullet\ \rightarrow_u\_closed\ s \Leftrightarrow \forall d\ c \bullet\ d \in_u s \wedge c \in_u s \Rightarrow d \rightarrow_u c \in_u s$

2.7 Functional Abstraction

Functional abstraction is defined as a new variable binding construct yeilding a functional set.

2.7.1 Abstraction

Because of the closeness to lambda abstraction λ_u is used as the name of a new binder for set theoretic functional abstraction.

SML

```
declare_binder "λu";
```

To define a functional set we need a HOL function over sets together with a set which is to be the domain of the function. Specification of the range is not needed. The binding therefore yields a function which maps sets to sets (maps the domain to the function).

The following definition is a placeholder, a more abstract definition might eventually be Substituted. The function is defined as that Subset of the cartesian product of the set s and its image under the function f which coincides with the graph of f over s.

HOL Constant

$\$\lambda_u : ({}'a\ GSU \rightarrow {}'a\ GSU) \rightarrow {}'a\ GSU \rightarrow {}'a\ GSU$

$\forall f\ s \bullet\ \$\lambda_u\ f\ s = Sep_u\ (s \times_u (Imagep_u\ f\ s))\ (\lambda p \bullet\ Snd_u\ p = f\ (Fst_u\ p))$

2.8 Application and Extensionality

In this section we define function application and show that functions are extensional.

2.8.1 Application

Application by juxtaposition cannot be overloaded and is used for application of HOL functions. Application of functional sets is therefore defined as an infix operator whose name is the empty name Subscripted by "u".

SML

```
declare_infix (250,"u");
```

The particular form shown here is innovative in the value specified for applications of functions to values outside their domain. The merit of the particular value chosen is that it makes true an extensionality theorem which quantifies over all sets as arguments to the function, which might not otherwise be the case. Whether this form is useful I don't know. Generally a result with fewer conditionals is harder to prove but easier to use, but in this case I'm not so sure of the benefit.

It may be noted that it may also be used to apply a non-functional relation, if what you want it some arbitrary value (selected by the choice function) to which some object relates.

HOL Constant

$\$_u : 'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

$$\begin{aligned}
&\forall f\ x \bullet\ f\ _u\ x = \\
&\quad if\ \exists y \bullet\ x \mapsto_u y \in_u f \\
&\quad then\ \epsilon y \bullet\ x \mapsto_u y \in_u f \\
&\quad else\ f
\end{aligned}$$

$$\begin{aligned}
&app_thm1 = \\
&\quad \vdash \forall f\ x \bullet\ (\exists_1 y \bullet\ x \mapsto_u y \in_u f) \\
&\quad\ \ \Rightarrow x \mapsto_u (f\ _u\ x) \in_u f
\end{aligned}$$

$$\begin{aligned}
&app_thm2 = \\
&\quad \vdash \forall f\ x\ y \bullet\ Fun_u\ f \wedge (x \mapsto_u y \in_u f) \\
&\quad\ \ \Rightarrow f\ _u\ x = y
\end{aligned}$$

$$\begin{aligned}
&app_thm3 = \\
&\quad \vdash \forall f\ x \bullet\ Fun_u\ f \wedge x \in_u Dom_u\ f \\
&\quad\ \ \Rightarrow x \mapsto_u f\ _u\ x \in_u f
\end{aligned}$$

$$\begin{aligned}
&o_{u}\text{-}_{u}\text{-}thm = \\
&\quad \vdash \forall f\ g\ x \bullet\ Fun_u\ f \wedge Fun_u\ g \wedge x \in_u Dom_u\ g \wedge Ran_u\ g \subseteq_u Dom_u\ f \\
&\quad\ \ \Rightarrow (f\ o_u\ g)\ _u\ x = f\ _u\ g\ _u\ x
\end{aligned}$$

$$app\_in\_Ran\_thm = \vdash \forall\ x\ i \bullet\ Fun_u\ i \wedge x \in_u Dom_u\ i \Rightarrow i\ _u\ x \in_u Ran_u\ i$$

2.8.2 The "Type" of an Application (1)

The following theorem states that the result of applying a partial function to a value in its domain is a value in its codomain.

SML

```
set_goal([],
	⌜∀f s t u• f ∈_u s ⇸_u t ∧
	u ∈_u Dom_u f ⇒
	f _u u ∈_u t⌝);
a (prove_tac[
	get_spec ⌜$⇸_u⌝,
	Dom_u_def]);
a (all_fc_tac [app_thm2] THEN asm_rewrite_tac[]);
a (all_fc_tac [f↦_us_thm1]);
a (all_fc_tac [∈_u↔_u_thm]);
a (POP_ASM_T ante_tac THEN asm_rewrite_tac []);
val _u∈_u_thm = save_pop_thm "_u∈_u_thm";
```

2.8.3 The "Type" of an Application (2)

The following theorem states that the result of applying a total function to a value in its domain is a value in its codomain.

2.8.4 Partial functions are total

Every partial function is total over its domain. (there is an ambiguity in the use of the term "domain" for a partial function. It might mean the left hand operand of some partial function space construction within which the partial function concerned may be found, or it might mean the set of values over which the function is defined. Here we are saying that if f is a partial function over A, then its domain is some Subset of A and f is a total function over that Subset of A.)

$\in_u\!\nrightarrow_u\!\Rightarrow\!\in_u\!\rightarrow_u\!\_thm$ =

$\quad\vdash \forall f\ s\ t\ u\bullet\ f \in_u s \nrightarrow_u t \Rightarrow f \in_u Dom_u\ f \rightarrow_u t$

2.9 The Identity Function

HOL Constant

$Id_u : 'a\ GSU \rightarrow 'a\ GSU$

$\forall s\bullet\ Id_u\ s = Sep_u$

$\quad (s \times_u s)$

$\quad \lambda x\bullet\ Fst_u\ x = Snd_u\ x$

$\boldsymbol{Id_u\_thm1}$ =
  $\vdash \forall s\ x\bullet\ x \in_u Id_u\ s \Leftrightarrow \exists y\bullet\ y \in_u s \wedge x = y \mapsto_u y$

$\boldsymbol{Id_u\_ap\_thm}$ =
  $\vdash \forall s\ x\bullet\ x \in_u s \Rightarrow (Id_u\ s)\ _u\ x = x$

$\boldsymbol{Id_u{\in_u}{\leftrightarrow_u}\_thm1}$ =
  $\vdash \forall s\ t\ u\bullet\ s \subseteq_u t \cap_u u \Rightarrow Id_u\ s \in_u t \leftrightarrow_u u$

$\boldsymbol{Id_u{\in_u}{\leftrightarrow_u}\_thm2}$ =
  $\vdash \forall s\ t\ u\bullet\ s \subseteq_u t \Rightarrow Id_u\ s \in_u t \leftrightarrow_u t$

$\boldsymbol{tc\_Id_u\_thm}$ = $\vdash \forall\ s\ t\bullet\ s \in_u t \Rightarrow s\ {\in_u}^+\ Id_u\ t$

$\boldsymbol{Id_u\_clauses}$ =
  $\vdash \forall s\bullet\ Rel_u(Id_u\ s) \wedge Fun_u\ (Id_u\ s) \wedge Dom_u(Id_u\ s) = s \wedge Ran_u(Id_u\ s) = s$

2.10 Override

Override is an operator over sets which is intended primarily for use with functions. It may be used to change the value of the function over any part of its domain by overriding it with a function which is defined only for those values.

SML

```
declare_infix (250,"⊕u");
```

HOL Constant

$\$\oplus_u : 'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall s\ t\bullet\ s \oplus_u t = Sep_u\ (s \cup_u t)$
  $\lambda x\bullet\ if\ Fst_u\ x \in_u Dom_u\ t\ then\ x \in_u t\ else\ x \in_u s$

$\boldsymbol{\in_u\oplus_u\_thm}$ =
  $\vdash \forall\ s\ t\ x\bullet\ x \in_u s \oplus_u t = (if\ Fst_u\ x \in_u Dom_u\ t\ then\ x \in_u t\ else\ x \in_u s)$

$\boldsymbol{\mapsto_u\in_u\oplus_u\_thm}$ =
  $\vdash \forall\ s\ t\ x\ y$
  $\bullet\ x \mapsto_u y \in_u s \oplus_u t = (x \mapsto_u y \in_u t \vee \neg\ x \in_u Dom_u\ t \wedge x \mapsto_u y \in_u s)$

$\boldsymbol{\oplus_u\_Rel_u\_thm}$ =
  $\vdash \forall\ s\ t\bullet\ Rel_u\ s \wedge Rel_u\ t \Rightarrow Rel_u\ (s \oplus_u t)$

$\boldsymbol{\oplus_u\_Fun_u\_thm}$ =
  $\vdash \forall\ s\ t\bullet\ Fun_u\ s \wedge Fun_u\ t \Rightarrow Fun_u\ (s \oplus_u t)$

2.11 Proof Contexts

Finalisation of a proof context.

2.11.1 Proof Context

SML

```
add_pc_thms "'gsu−fun" ([
        Field_u_∅_u_thm,
        ∅_u∈_u↛_u_thm]);
add_rw_thms [Fun_u_∅_u_thm] "'gsu−fun";
add_sc_thms [Fun_u_∅_u_thm] "'gsu−fun";
set_merge_pcs ["basic_hol", "'gsu−ax", "'gsu−fun"];
commit_pc "'gsu−fun";
```

3 Ordinals

A new "gsu-ord" theory is created as a child of "gsu-ax". The theory will contain the definitions of ordinals and related material for general use. I began by roughly following "Set Theory" by Frank Drake, chapter 2 section 2, but later development was undertaken without access to that book and was driven by applications.

3.0.2 Motivation

When I first started this theory of Ordinals I was motivated by interest and self-education, though the set theory in the context of which the development was undertaken had been initiated for my foundational research. The educational interest didn't get me very far, and this theory languished scarcely begun for many years. The set theory continued to be used for my foundational research, which also has progressed very slowly and did not begin to make demands on the theory of ordingals until late in 2012. Even then, the things I need are very modest, to know for example, that ordinal addition is associative.

Some of the material required is not specific to set theory and is quite widely applicable (in which case I actually develop it elsewhere and then just use it here). Well-foundedness and induction over well-founded relations is the obvious case relevant to this part of Drake. The recursion theorem is the important more general result which appears in the next section in Drake. "more general" means "can be developed as a polymorphic theory in HOL and applied outside the context of set theory". In fact these things have to be developed in the more general context to be used in the ways they are required in the development of set theory, since, for example, one wants to do definitions by recursion over the set membership relation where neither the function defined nor the relevant well-founded relation are actually sets.

3.0.3 Divergence

I never did follow Drake slavishly, and I no longer have a copy of his book, but now refer to Jech as necessary. However, the level remains elementary, so this is rarely necessary.

Sometimes the context in which I am doing the work makes some divergence desirable or necessary. For example, I am doing the work in the context of a slightly eccentric set theory ("Galactic Set

Theory") which mainly makes no difference, but has a non-standard formulation of the axiom of foundation. Mainly this is covered by deriving the standard formulation and its consequences and using them where this is used by Drake (in proving the trichotomy theorem). However, the machinery for dealing with well-foundedness makes a difference to how induction principles are best formulated and derived.

Sometimes I looked at what he had done and I think, "no way am I going to do that". Not necessarily big things, for example, I couldn't use his definition of successor ordinal which he pretty much admits himself is what I would call a kludge. If I started again I would not use his definition of ordinal either, I would use an inductive definition (nicer in a higher order theory).

3.0.4 The Theory ord

The new theory is first created, together with a proof context which we will build up as we develop the theory.

SML

```
open_theory "gsu-ax";
force_new_theory "gsu-ord";
(* new_parent "wf_recp"; *)
force_new_pc "'gsu-ord";
merge_pcs ["'savedthm_cs_∃_proof"] "'gsu-ord";
set_merge_pcs ["basic_hol", "'gsu-ax", "'gsu-ord"];
```

3.1 Definition of Ordinal

An ordinal is defined as a transitive and connected set. The usual ordering over the ordinals is defined and also the successor function.

3.1.1 The Definition

The concept of transitive set has already been defined in theory *gsu-ax*. The concepts *connected* and *ordinal* are now defined.

The possible presence of urelements causes complications here, we have to ensure that ordinals are hereditarily sets. For this is does not suffice to require that every ordinal is a set, for we do not assert but must prove from the definition that all members of an ordinal are ordinals.

It does suffice to insist that an ordinal is a set and that all its members are sets, so we define that property here.

HOL Constant

$\boldsymbol{SetOfSets_u} : {'a}\ GSU \rightarrow BOOL$

$\forall s :'a\ GSU\bullet\ SetOfSets_u\ s \Leftrightarrow Set_u\ s \land \forall t:'a\ GSU\bullet\ t \in_u s \Rightarrow Set_u\ t$

HOL Constant

$\boldsymbol{Connected_u} : 'a\ GSU \to BOOL$

$\forall s :'a\ GSU \bullet\ Connected_u\ s \Leftrightarrow$
$\quad \forall t\ u :'a\ GSU \bullet\ t \in_u s \land u \in_u s \Rightarrow t \in_u u \lor t = u \lor u \in_u t$

HOL Constant

$\boldsymbol{Ordinal_u} : 'a\ GSU \to BOOL$

$\forall s :'a\ GSU \bullet\ Ordinal_u\ s \Leftrightarrow SetOfSets_u\ s \land Transitive_u\ s \land Connected_u\ s$

$\boldsymbol{Set_u\_ord_u\_thm} = \quad \vdash \forall s \bullet\ Ordinal_u\ s \Rightarrow Set_u\ s$
$\boldsymbol{gsu\_ordinal\_ext\_thm} = \quad \vdash \forall\ s\ t \bullet\ Ordinal_u\ s \land Ordinal_u\ t$
$\quad \Rightarrow (s = t \Leftrightarrow (\forall\ e \bullet\ e \in_u s \Leftrightarrow e \in_u t))$
$\boldsymbol{tc\in_u\_ord_u\_thm} = \quad \vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land \beta \in_u{}^+ \alpha \Rightarrow \beta \in_u \alpha$

We now introduce infix ordering relations over ordinals.

SML

```
declare_infix(240,"<u");
declare_infix(240,"≤u");
```

HOL Constant

$\$<_u : 'a\ GSU \to 'a\ GSU \to BOOL$

$\forall x\ y:'a\ GSU \bullet\ x <_u y \Leftrightarrow Ordinal_u\ x \land Ordinal_u\ y \land x \in_u y$

$\boldsymbol{lt_u\_\in_u\_thm} =$
$\quad \vdash \forall\ \alpha\ \beta \bullet\ \alpha <_u \beta \Rightarrow Ordinal_u\ \alpha \land Ordinal_u\ \beta \land \alpha \in_u \beta$

$\boldsymbol{\in_u\_lt_u\_thm} =$
$\quad \vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \land \alpha \in_u \beta \Rightarrow \alpha <_u \beta$

$\boldsymbol{\in_u\_\Leftrightarrow\_lt_u\_thm} =$
$\quad \vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow (\alpha \in_u \beta \Leftrightarrow \alpha <_u \beta)$

$\boldsymbol{tc\in_u\_\Leftrightarrow\_lt_u\_thm} =$
$\quad \vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow (\alpha \in_u{}^+ \beta \Leftrightarrow \alpha <_u \beta)$

$\boldsymbol{ord_u\_\in_u\_\subset_u\_thm} =$
$\quad \vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow \beta \subset_u \alpha)$

$\boldsymbol{ord_u\_\in_u\_trans\_thm} =$
$\quad \vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow (\forall\ \beta\ \gamma \bullet\ \beta \in_u \alpha \land \gamma \in_u \beta \Rightarrow \beta \in_u \alpha)$

$\boldsymbol{lt_u\_\subset_u\_thm}$ =
$\vdash \forall\ \alpha\ \beta \bullet\ \alpha <_u \beta \Rightarrow \alpha \subset_u \beta$

$\boldsymbol{lt_u\_trans\_thm}$ =
$\vdash \forall\ \alpha\ \beta\ \gamma \bullet\ \alpha <_u \beta \land \beta <_u \gamma \Rightarrow \alpha <_u \gamma$

HOL Constant

$\$\leqslant_u : 'a\ GSU \to 'a\ GSU \to BOOL$

$\forall x\ y{:}'a\ GSU \bullet\ x \leqslant_u y \Leftrightarrow Ordinal_u\ x \land Ordinal_u\ y \land (x \in_u y \lor x = y)$

$\boldsymbol{\leqslant_u\_lt_u\_thm}$ =
$\vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow (\alpha \leqslant_u \beta \Leftrightarrow \alpha <_u \beta \lor \alpha = \beta)$

$\boldsymbol{\leqslant_u\_lt_u\_thm2}$ =
$\vdash \forall\ x\ y \bullet\ x \leqslant_u y \Leftrightarrow Ordinal_u\ x \land Ordinal_u\ y \land (x <_u y \lor x = y)$

$\boldsymbol{\leqslant_u\_\subseteq_u\_thm}$ =
$\vdash \forall\ \alpha\ \beta \bullet\ \alpha \leqslant_u \beta \Rightarrow \alpha \subseteq_u \beta$

$\boldsymbol{\leqslant_u\_trans\_thm}$ =
$\vdash \forall\ \alpha\ \beta\ \gamma \bullet\ \alpha \leqslant_u \beta \land \beta \leqslant_u \gamma \Rightarrow \alpha \leqslant_u \gamma$

$\boldsymbol{\leqslant_u\_lt_u\_trans\_thm}$ =
$\vdash \forall\ \alpha\ \beta\ \gamma \bullet\ \alpha \leqslant_u \beta \land \beta <_u \gamma \Rightarrow \alpha <_u \gamma$

$\boldsymbol{lt_u\_\leqslant_u\_trans\_thm}$ =
$\vdash \forall\ \alpha\ \beta\ \gamma \bullet\ \alpha <_u \beta \land \beta \leqslant_u \gamma \Rightarrow \alpha <_u \gamma$

The following definition gives the successor function over the ordinals (this appears later in Drake).

HOL Constant

$\boldsymbol{Suc_u} : 'a\ GSU \to 'a\ GSU$

$\forall x{:}'a\ GSU \bullet\ Suc_u\ x = x \cup_u (Unit_u\ x)$

$\boldsymbol{Set_u\_Suc_u\_thm}$ = $\vdash \forall\ s \bullet\ Set_u\ (Suc_u\ s)$
$\boldsymbol{\subseteq_u\_Suc_u\_thm}$ = $\vdash \forall\ s \bullet\ s \subseteq_u Suc_u\ s \land Unit_u\ s \subseteq_u Suc_u\ s$
$\boldsymbol{\in_u\_Suc_u\_thm}$ = $\vdash \forall x\ y \bullet\ x \in_u Suc_u\ y \Leftrightarrow x \in_u y \lor x = y$
$\boldsymbol{\in_u\_Suc_u\_thm2}$ = $\vdash \forall\ s \bullet\ s \in_u Suc_u\ s$

3.2 The Empty Set and its Successors

We prove that the empty set is an ordinal, and that the members of an ordinal and the successor of an ordinal are ordinals.

3.2.1 The Empty Set is an Ordinal

First we prove that the empty set is an ordinal, which requires only rewriting with the relevant definitions. We then give a theorem necessitated by the presence of urelements which are extensionally equivalent to the empty set, telling us that the only ordinal with empty extension is the empty set.

$\boldsymbol{ord_u\_\varnothing_u\_thm} = \quad \vdash Ordinal_u\ \varnothing_u$
$\boldsymbol{ord_u\_eq\_\varnothing_u\_thm} = \ \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \land \alpha =_u \varnothing_u \Rightarrow \alpha = \varnothing_u$

3.2.2 The Successor of an Ordinal is an Ordinal

Next we prove that the successor of an ordinal is an ordinal. This is done in two parts, transitivity and connectedness.

$\boldsymbol{tran_u\_suc_u\_tran_u\_thm} = \vdash \forall x\bullet\ Transitive_u\ x \Rightarrow Transitive_u\ (Suc_u\ x)$
$\boldsymbol{conn_u\_suc_u\_conn_u\_thm} = \vdash \forall x\bullet\ Connected_u\ x \Rightarrow Connected_u\ (Suc_u\ x)$

These together enable us to prove:

$\boldsymbol{ord_u\_suc_u\_ord_u\_thm} = \quad \vdash \forall\ x\bullet\ Ordinal_u\ x \Rightarrow Ordinal_u\ (Suc_u\ x)$

3.2.3 The Ordinal Zero is not a Successor

$\boldsymbol{\varnothing_u\_not\_Suc_u\_thm} = \vdash \neg\ (\exists\ \alpha\bullet\ Suc_u\ \alpha = \varnothing_u)$
$\boldsymbol{\neg\_eq\_suc_u\_thm} = \quad \vdash \forall\ \alpha\bullet\ \neg\ \alpha = Suc_u\ \alpha$
$\boldsymbol{\leqslant_u\_suc_u\_thm} = \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow \alpha \leqslant_u Suc_u\ \alpha$
$\boldsymbol{lt_u\_suc_u\_thm} = \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow \alpha <_u Suc_u\ \alpha$

3.2.4 The members of an Ordinal are Ordinals

We now aim to prove that the members of an ordinal are ordinals. We do this by proving first that they are connected and then that they are transitive. First however, we show that any Subset of a connected set is connected.

$\boldsymbol{conn\_\subseteq_u\_conn} = \forall x\bullet\ Connected_u\ x \Rightarrow \forall\ y{:}'a\ GSU\bullet\ y \subseteq_u x \Rightarrow Connected_u\ y$

Now we show that any member of an ordinal is an ordinal.

$\boldsymbol{conn\_\in_u\_ord} = \forall\ x\bullet\ Ordinal_u\ x \Rightarrow \forall\ y{:}'a\ GSU\bullet\ y \in_u x \Rightarrow Connected_u\ y$

To prove that the members of an ordinal are transitive, well-foundedness is needed. Now we are ready to prove that the members of an ordinal are transitive.

$\boldsymbol{tran_u\_\in_u\_ord} = \vdash \forall\ x\bullet\ Ordinal_u\ x \Rightarrow \forall\ y{:}'a\ GSU\bullet\ y \in_u x \Rightarrow Transitive_u\ y$

We also need to prove that the members of an ordinal are all sets of sets.

$\boldsymbol{setosets\_\in_u\_ord} = \vdash \forall\ x\bullet\ Ordinal_u\ x \Rightarrow (\forall\ y\bullet\ y \in_u x \Rightarrow SetOfSets_u\ y)$

Finally we prove that all members of an ordinal are ordinals.

$\boldsymbol{ord_u\_\in_u\_ord_u\_thm} = \vdash \forall\ x\bullet\ Ordinal_u\ x \Rightarrow \forall\ y{:}'a\ GSU\bullet\ y \in_u x \Rightarrow Ordinal_u\ y$
$\boldsymbol{ord_u\_tc\in_u\_ord_u\_thm} = \vdash \forall\ x\bullet\ Ordinal_u\ x \Rightarrow (\forall\ y\bullet\ y \in_u{}^+ x \Rightarrow Ordinal_u\ y)$

I think things might be simpler if we had an extensionality principle expressed in terms of this ordering, and then characterise addition using the ordering.

$\boldsymbol{ord_u\_ext\_thm} =$
$\vdash \forall\ \alpha\ \beta\bullet\ Ordinal_u\ \alpha \wedge Ordinal_u\ \beta \Rightarrow (\alpha = \beta \Leftrightarrow (\forall\ \gamma\bullet\ \gamma <_u \alpha \Leftrightarrow \gamma <_u \beta))$

3.2.5 Galaxies are Closed under Suc

$GClose_u\_Suc_u\_thm = \vdash \forall g\bullet\ galaxy_u\ g \Rightarrow \forall x\bullet\ x \in_u g \Rightarrow Suc_u\ x \in_u g$

3.3 Ordinals are Linearly Ordered

We prove that the ordinals are linearly ordered by $<_u$.

First we prove some lemmas:

$\boldsymbol{tran_u\_\cap_u\_thm} = \vdash \forall\ x\ y\bullet\ Transitive_u\ x \wedge Transitive_u\ y \Rightarrow Transitive_u\ (x \cap_u y)$
$\boldsymbol{tran_u\_\cup_u\_thm} = \vdash \forall\ x\ y\bullet\ Transitive_u\ x \wedge Transitive_u\ y \Rightarrow Transitive_u\ (x \cup_u y)$
$\boldsymbol{\varnothing_u\_\leqslant_u\_thm} = \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow \varnothing_u \leqslant_u \alpha$
$\boldsymbol{\varnothing_u\_eq\_lt_u\_thm} = \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow \alpha = \varnothing_u \vee \varnothing_u <_u \alpha$
$\boldsymbol{\varnothing_u\_neq\_\in_u\_thm} = \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \wedge \neg\ \alpha = \varnothing_u \Rightarrow \varnothing_u \in_u \alpha$
$\boldsymbol{\varnothing_u\_neq_u\_\in_u\_thm} = \vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \wedge \neg\ \alpha =_u \varnothing_u \Rightarrow \varnothing_u \in_u \alpha$

$\boldsymbol{conn\_\cap_u\_thm} =$
$\vdash \forall\ x\ y{:}'a\ GSU\bullet\ Connected_u\ x \wedge Connected_u\ y \Rightarrow Connected_u\ (x \cap_u y)$
$\boldsymbol{setofsets\_\cap_u\_thm} =$
$\vdash \forall\ x\ y\bullet\ SetOfSets_u\ x \wedge SetOfSets_u\ y \Rightarrow SetOfSets_u\ (x \cap_u y)$
$\boldsymbol{ord_u\_\cap_u\_thm} =$
$\vdash \forall\ x\ y{:}'a\ GSU\bullet\ Ordinal_u\ x \wedge Ordinal_u\ y \Rightarrow Ordinal_u\ (x \cap_u y)$

$\boldsymbol{trichot_u\_lemma} =$
$\vdash \forall\ x\ y\bullet\ Ordinal_u\ x \wedge Ordinal_u\ y \wedge x \subseteq_u y \wedge \neg\ x = y \Rightarrow x \in_u y$

$\textbf{\textit{trich\_for\_ord}}_{\textbf{\textit{u}}}\textbf{\textit{\_thm}}$ =

$\vdash \forall\ x\ y \bullet\ Ordinal_u\ x \land Ordinal_u\ y \Rightarrow x <_u y \lor x = y \lor y <_u x$

$\subseteq_{\boldsymbol{u}}\_\leqslant_{\boldsymbol{u}}\textbf{\textit{\_thm}}$ =

$\vdash \forall\ x\ y \bullet\ Ordinal_u\ x \land Ordinal_u\ y \Rightarrow (x \subseteq_u y \Leftrightarrow x \leqslant_u y)$

$\subseteq_{\boldsymbol{u}}\_\leqslant_{\boldsymbol{u}}\textbf{\textit{\_thm1}}$ =

$\vdash \forall\ x\ y \bullet\ Ordinal_u\ x \land Ordinal_u\ y \land x \subseteq_u y \Rightarrow x \leqslant_u y$

$\textbf{\textit{lt}}_{\boldsymbol{u}}\_\Leftrightarrow\_\subset_{\boldsymbol{u}}\textbf{\textit{\_thm}}$ =

$\vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow (\alpha <_u \beta \Leftrightarrow \alpha \subset_u \beta)$

$\subset_{\boldsymbol{u}}\_\Leftrightarrow\_\textbf{\textit{lt}}_{\boldsymbol{u}}\textbf{\textit{\_thm}}$ =

$\vdash \forall\ x\ y \bullet\ Ordinal_u\ x \land Ordinal_u\ y \Rightarrow (x \subset_u y \Leftrightarrow x <_u y)$

$\subset_{\boldsymbol{u}}\_\textbf{\textit{lt}}_{\boldsymbol{u}}\textbf{\textit{\_thm1}}$ =

$\vdash \forall\ x\ y \bullet\ Ordinal_u\ x \land Ordinal_u\ y \land x \subset_u y \Rightarrow x <_u y$

$\textbf{\textit{ord}}_{\boldsymbol{u}}\_\textbf{\textit{sub}}_{\boldsymbol{u}}\_\in_{\boldsymbol{u}}\textbf{\textit{\_thm}}$ =

$\vdash \forall\ x\ y \bullet\ Ordinal_u\ x \land Ordinal_u\ y \land x \subset_u y \Rightarrow x \in_u y$

3.4 Successor and Limit Ordinals

Successor and limit ordinals are defined. Natural numbers are defined.

These definitions are not the ones used by Drake, and not only the names but the concepts differ. My successor predicate does not hold of the empty set. I use the name "natural number" where he talks of integers, and generally I'm chosing longer names.

HOL Constant

$\textbf{\textit{Successor}}_{\boldsymbol{u}} : 'a\ GSU \to BOOL$

$\forall s :'a\ GSU \bullet\ Successor_u\ s \Leftrightarrow \exists t \bullet\ Ordinal_u\ t \land s = Suc_u\ t$

HOL Constant

$\textbf{\textit{LimitOrdinal}}_{\boldsymbol{u}} : 'a\ GSU \to BOOL$

$\forall s :'a\ GSU \bullet\ LimitOrdinal_u\ s \Leftrightarrow Ordinal_u\ s \land \neg\ Successor_u\ s \land \neg\ s = \varnothing_u$

3.5 Induction

Induction theorems over ordinals.

$\textbf{\textit{Successor}}_{\boldsymbol{u}}\_\textbf{\textit{ord}}_{\boldsymbol{u}}\textbf{\textit{\_thm}}$ = $\vdash \forall x \bullet\ Successor_u\ x \Rightarrow Ordinal_u\ x$

3.5.1 Well-foundedness of the ordinals

First we prove that $<_u$ is well-founded.

$\textbf{\textit{wf\_lt}}_{\boldsymbol{u}}\textbf{\textit{\_thm}}$ = $\vdash well\_founded\ \$<_u$

Which yeild the following induction tactics:

SML

```
val 'a ORDINAL_INDUCTION_T = WF_INDUCTION_T wf_lt_u_thm;
val ordinal_induction_tac = wf_induction_tac wf_lt_u_thm;
```

3.5.2 Continuity

HOL Constant

$$\mathbf{Continuous_u} : ('a\ GSU \to 'a\ GSU) \to BOOL$$

$$\forall f \bullet\ Continuous_u\ f \Leftrightarrow \forall x \bullet\ LimitOrdinal_u\ x \Rightarrow f\ x = ClIm_u\ f\ x$$

3.5.3 An Ordinal is Zero, a successor or a limit

$$\mathbf{ord_u\_kind\_thm} = \vdash \forall n \bullet\ Ordinal_u\ n \Rightarrow n = \varnothing_u \lor Successor_u\ n \lor LimitOrdinal_u\ n$$

3.6 Supremum and Strict Supremum

The supremum of a set of ordinals is the smallest ordinal greater than or equal to every ordinal in the set. With the Von Neumann representation of ordinals this is just the union of the set of ordinals.

SML

```
declare_infix (200, "Ub_u");
declare_infix (200, "Sub_u");
```

HOL Constant

$$\$\mathbf{Ub_u} : 'a\ GSU \to 'a\ GSU \to BOOL$$

$$\forall \alpha\ \beta \bullet\ \alpha\ Ub_u\ \beta \Leftrightarrow \forall \gamma \bullet\ \gamma \in_u \alpha \Rightarrow \gamma \leqslant_u \beta$$

HOL Constant

$$\mathbf{Sup_u} : 'a\ GSU \to 'a\ GSU$$

$$\forall \alpha \bullet\ Sup_u\ \alpha = \bigcup_u \alpha$$

$$\mathbf{ord_u\_limit\_thm} = \vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (\bigcup_u \alpha)$$

$$\mathbf{ord_u\_Sup_u\_thm} = \vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (Sup_u\ \alpha)$$

$$\mathbf{Sup_u\_lUb_u\_thm} = \vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow \alpha\ Ub_u\ Sup_u\ \alpha \land (\forall\ \gamma \bullet\ Ordinal_u\ \gamma \land \alpha\ Ub_u\ \gamma \Rightarrow Sup_u\ \alpha \leqslant_u \gamma)$$

The operand here is intended to be an arbitrary set of ordinals and the result is the smallest ordinal strictly greater than any in the set.

HOL Constant

$\$\boldsymbol{Sub_u} : {}'a\ GSU \to {}'a\ GSU \to BOOL$

$\forall \alpha\ \beta \bullet\ \alpha\ Sub_u\ \beta \Leftrightarrow \forall \gamma \bullet\ \gamma \in_u \alpha \Rightarrow \gamma <_u \beta$

HOL Constant

$\boldsymbol{Ssup_u} : {}'a\ GSU \to {}'a\ GSU$

$\forall \alpha \bullet\ Ssup_u\ \alpha = \bigcup_u (Imagep_u\ Suc_u\ \alpha)$

$\boldsymbol{Ssup_u\_ord_u\_thm}$ =
$\vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (Ssup_u\ \alpha)$
$\boldsymbol{Ssup_u\_\in_u\_thm}$ =
$\vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow \beta \in_u Ssup_u\ \alpha)$
$\boldsymbol{Ssup_u\_lt_u\_thm}$ =
$\vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow \beta <_u Ssup_u\ \alpha)$
$\boldsymbol{Ssup_u\_\subseteq_u\_thm}$ =
$\vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow \alpha\ Sub_u\ Ssup_u\ \alpha$
$\boldsymbol{Ssup_u\_TrCl_u\_thm}$ =
$\vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ssup_u\ \alpha = TrCl_u\ \alpha$
$\boldsymbol{TrCl_u\_ord_u\_thm}$ =
$\vdash \forall\ \alpha \bullet\ (\forall\ \beta \bullet\ \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (TrCl_u\ \alpha)$

I defined this "strict supremum" for use in defining the arithmetic operators over ordinals by transfinite recursion. Later I realised that strict supremum is just the same as transitive closure if the operand is a set of ordinals, and decided that it would be nice to have a new form of the replacement principle, which resulted in the *ClIm* function for defining operations over ordinals. So I think now that strict supremum will prove to be redundant.

3.7 Rank

We define the rank of a set.

3.7.1 The Consistency Proof

This is a hangover from the days before I started to use automatic proofs based on recursion principles.

Before introducing the definition of rank we undertake the proof necessary to establish that the definition is conservative. The key lemma in this proof is the proof that the relevant functional "respects" the membership relation.

$\boldsymbol{respect_u\_lemma}$ =
$\vdash (\lambda\ f\ x \bullet\ \bigcup_u\ (Imagep_u\ (Suc_u\ o\ f)\ x))\ respects\ \$\in_u$

Armed with that lemma we can now prove that the function which we will call "rank" exists (proof not shown).

HOL Constant

$\boldsymbol{Rank_u} : 'a\ GSU \to 'a\ GSU$

$\forall x \bullet\ Rank_u\ x = \bigcup_u\ (Imagep_u\ (Suc_u\ o\ Rank_u)\ x)$

3.8 Ordinal Arithmetic

Arithmetic operators are defined by transfinite induction. Some machinery for this has already been provided, the principle tool being the operator *ClIm* giving the transitive closure of the image of a set under a function. In the inductive definitions the base case will be for the ordinal zero and will be given explicitly, the induction case will be obtained using *ClIm* to obtain the closure of the set of values obtained by a function from the predecessors of the ordinal in question (i.e. its members).

If the function used in the definition is "closure compatible" (*ClCo*) then an associative law for the operation can be obtained using general properties of the *ClIm* operator. It is therefore helpful to begin with some ordinal specific results about the *ClIm* operator.

3.8.1 Addition

The difficulty of reasoning of course depends much upon the definition chosen. I have tried more than one!

SML

```
declare_infix (300, "+_u");
declare_infix (300, "−−_u");
```

HOL Constant

$\$+_{\boldsymbol{u}} : 'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall \alpha\ \beta \bullet\ \alpha +_u \beta = if\ \beta =_u \varnothing_u\ then\ set_u\ \alpha\ else\ ClIm_u\ (\$+_u\ \alpha)\ \beta$

$\boldsymbol{Set_u\_plus_u\_thm} =$
$\quad \vdash \forall\ \alpha\ \beta \bullet\ Set_u\ (\alpha +_u \beta)$
$\boldsymbol{ord_u\_set_u\_thm} =$
$\quad \vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow set_u\ \alpha = \alpha$
$\boldsymbol{ord_u\_eq_u\_\varnothing_u\_thm} =$
$\quad \vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \land \alpha =_u \varnothing_u \Rightarrow \alpha = \varnothing_u$
$\boldsymbol{ord_u\_plus_u\_thm} =$
$\quad \vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow Ordinal_u\ (\alpha +_u \beta)$
$\boldsymbol{\varnothing_u\_plus_u\_thm} =$
$\quad \vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow \varnothing_u +_u \alpha = \alpha$
$\boldsymbol{plus_u\_\varnothing_u\_thm} =$
$\quad \vdash \forall\ \alpha \bullet\ Set_u\ \alpha \Rightarrow \alpha +_u \varnothing_u = \alpha$
$\boldsymbol{plus_u\_\varnothing_u\_thm2} =$

$\vdash \forall\ \alpha\bullet\ \alpha +_u \varnothing_u =_u \alpha$

$\boldsymbol{plus_u\_\varnothing_u\_thm3}$ =

$\vdash \forall\ \alpha\bullet\ \alpha +_u \varnothing_u = set_u\ \alpha$

$\boldsymbol{plus_u\_\varnothing_u\_thm4}$ =

$\vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow \alpha +_u \varnothing_u = \alpha$

$\boldsymbol{plus_u\_ur\_thm}$ =

$\vdash \forall\ \alpha\ x\bullet\ Set_u\ \alpha \land x =_u \varnothing_u \Rightarrow \alpha +_u x = \alpha$

$\boldsymbol{plus_u\_ur\_thm2}$ =

$\vdash \forall\ \alpha\ x\bullet\ x =_u \varnothing_u \Rightarrow \alpha +_u x =_u \alpha$

$\boldsymbol{plus_u\_ur\_thm3}$ =

$\vdash \forall\ \alpha\ x\bullet\ x =_u \varnothing_u \Rightarrow \alpha +_u x =_u set_u\ \alpha$

$\boldsymbol{ClCo_u\_plus_u\_thm}$ =

$\vdash \forall\ \alpha\bullet\ ClCo_u\ (\$+_u\ \alpha)$

$\boldsymbol{plus_u\_eq\_\varnothing_u\_thm}$ =

$\vdash \forall\ \alpha\ \beta\bullet\ \alpha +_u \beta =_u \varnothing_u \Rightarrow \alpha =_u \varnothing_u \land \beta =_u \varnothing_u$

$\boldsymbol{\in_u\_ClIm_u\_plus_u\_thm}$ =

$\vdash \forall\ x\ \alpha\ t\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ t \Rightarrow$
$(x \in_u ClIm_u\ (\$+_u\ \alpha)\ t$
$\Leftrightarrow (\exists\ y\bullet\ y \in_u t \land (x = \alpha +_u y \lor x \in_u \alpha +_u y)))$

3.8.2 Another Experiment

The number of "cases" of course makes a big difference to the complexity of proofs. I pondered for a while about whether the conditional on emptyness of one operand in my definition of addition could be eliminated, and eventually came up with a scheme.

Rather than reworking lots of material with a new definition, I think it will be easy to prove the new definition equivalent to the old, so that I get to use either.

However, only a qualified equality is obtainable, which is not quite as simple to apply.

$\boldsymbol{plus_u\_def\_thm}$ =

$\vdash \forall\ \alpha\ \beta\bullet\ Ordinal_u\ \beta \Rightarrow \alpha +_u \beta = \alpha \cup_u ClIm_u\ (\$+_u\ \alpha)\ \beta$

This does reduce the number of explicit case splits, but the union results in automatic case splits, and the associativity proof still has 10 cases. Nevertheless, it does seem to be significantly easier using this characterisation of addition.

The problem with this characterisation is that it does not come out very nicely if you use it in an extensional proof. It expands into the use of both membership and the transitive closure of membership, both of which in the present context are really just the ordering relation on the ordinals.

Since we have an extensionality principle expressed in terms of the ordering on the ordinals, it might be useful to have an extensional characterisation of addition in the same spirit.

To achieve this it is desirable to define the property of (HOL) functions over sets that they map ordinals to ordinals.

HOL Constant

$$\$\boldsymbol{OrdMap_u} : ('a\ GSU \rightarrow 'a\ GSU) \rightarrow BOOL$$

$$\forall f\bullet\ OrdMap_u\ f \Leftrightarrow \forall \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow Ordinal_u\ (f\ \alpha)$$

We can then prove the following sufficient conditions for $ClIm_u$ to deliver an ordinal:

$\boldsymbol{ClIm_u\_ord_u\_thm}$ =

$$\vdash \forall\ f\ \alpha\bullet\ OrdMap_u\ f\ \wedge\ Ordinal_u\ \alpha \Rightarrow Ordinal_u\ (ClIm_u\ f\ \alpha)$$

The required characterisaton of $ClIm_u$ in terms of the ordering is then:

$\boldsymbol{lt_u\_ClIm_u\_thm}$ =

$$\vdash \forall\ \alpha\ f\ \beta\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta\ \wedge\ OrdMap_u\ f$$
$$\Rightarrow (\alpha <_u ClIm_u\ f\ \beta \Leftrightarrow (\exists\ \delta\bullet\ \delta <_u \beta\ \wedge\ \alpha \leqslant_u f\ \delta))$$

Here are some examples of $OrdMap$s.

$\boldsymbol{OrdMap_u\_plus_u\_thm}$ = $\vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow OrdMap_u\ (\$+_u\ \alpha)$

A further characterisation of ordinal addition in terms of the ordering relation is then obtained. This is a further candidate for the definition. If it could be used as the definition there would be a saving in proof effort, so some consideration of whether this pattern of definition is more generally useful and could be supported by automatic consistency proof might be a good idea.

$\boldsymbol{plus_u\_def\_thm2}$ =

$$\vdash \forall\ \beta\ \alpha\ \gamma\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta \Rightarrow$$
$$(\gamma <_u \alpha +_u \beta \Leftrightarrow \gamma <_u \alpha\ \vee\ (\exists\ \delta\bullet\ \delta <_u \beta\ \wedge\ \gamma = \alpha +_u \delta))$$

$\boldsymbol{plus_u\_Suc_u\_thm}$ =

$$\vdash \forall\ \alpha\ \beta\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta \Rightarrow \alpha +_u Suc_u\ \beta = Suc_u\ (\alpha +_u \beta)$$

This extensionality principle suggests a style of proof based on these inequalities, so it is useful to have more results about inequalities.

$\boldsymbol{plus_u\_mono\_right\_thm}$ =

$$\vdash \forall\ \alpha\ \beta\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta \Rightarrow \alpha \leqslant_u \alpha +_u \beta$$

$\boldsymbol{plus_u\_mono\_right\_thm2}$ =

$$\vdash \forall\ \alpha\ \beta\ \gamma\bullet\ Ordinal_u\ \gamma\ \wedge\ \alpha <_u \beta \Rightarrow \gamma +_u \alpha <_u \gamma +_u \beta$$

Which help in the proof of associativity of addition:

$\boldsymbol{plus_u\_assoc\_thm}$ =

$$\vdash \forall\ \gamma\ \alpha\ \beta\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta\ \wedge\ Ordinal_u\ \gamma$$
$$\Rightarrow \alpha +_u \beta +_u \gamma = (\alpha +_u \beta) +_u \gamma$$

3.8.3 Subtraction

The following definition is of reverse subtraction, i.e. the right operand is subtracted from the left and is taken from the left of that operand so that the following lemma (as yet unproven) obtains:

desideratum

$\boldsymbol{--_u\_lemma} =$
$\quad \forall \alpha\ \beta \bullet\ \alpha \leqslant_u \beta \Rightarrow \alpha +_u (\beta --_u \alpha) = \beta$

HOL Constant

$\$--_{\boldsymbol{u}} : {'a}\ GSU \rightarrow {'a}\ GSU \rightarrow {'a}\ GSU$

T

3.9 Proof Context

In this section we define a proof context for ordinals.

3.9.1 Proof Context

Since almost everything is conditional there are few additional theorems to include:

$Set_u\_Suc_u\_thm,\ \subseteq_u\_Suc_u\_thm,\ \in_u\_Suc_u\_thm,\ Set_u\_ord_u\_thm,$
$Set_u\_plus_u\_thm,\ plus_u\_\varnothing_u\_thm2,\ ord_u\_\varnothing_u\_thm$

SML

```
commit_pc "'gsu−ord";
```

4 Natural Numbers

SML

```
open_theory "gsu−ord";
force_new_theory "gsu−nat";
force_new_pc "'gsu−nat";
merge_pcs ["'savedthm_cs_∃_proof"] "'gsu−nat";
set_merge_pcs ["basic_hol", "'gsu−ax", "'gsu−ord", "'gsu−nat"];
```

HOL Constant

$\boldsymbol{natural\_number_u} : {'a}\ GSU \rightarrow BOOL$

$\forall s :{'a}\ GSU \bullet\ natural\_number_u\ s \Leftrightarrow s = \varnothing_u \lor (Successor_u\ s \land \forall t \bullet\ t \in_u s \Rightarrow t = \varnothing_u \lor Successor_u\ t)$

4.0.2 Ordering the Natural Numbers

To get an induction principle for the natural numbers we first define a well-founded ordering over them. Since I don't plan to use this a lot I use the name $<_{un}$ (less than over the natural numbers defined in galactic set theory).

SML

```
declare_infix(240,"<_un");
```

HOL Constant

```
$<_un : 'a GSU → 'a GSU → BOOL
────────────────────────────
∀x y:'a GSU• x <_un y ⇔ natural_number_u x ∧ natural_number_u y ∧ x ∈_u y
```

Now we try to find a better proof that the one above that this is well-founded. And fail, this is just a more compact rendition of the same proof.

SML

```
set_goal([],⌜well_founded $<_un⌝);
a (asm_tac gsu_wf_thm1);
a (fc_tac [wf_restrict_wf_thm]);
a (SPEC_NTH_ASM_T 1 ⌜λx y:'a GSU• natural_number_u x ∧ natural_number_u y⌝ ante_tac
      THEN rewrite_tac[]);
a (lemma_tac ⌜$<_un = (λ x y:'a GSU• (natural_number_u x ∧ natural_number_u y) ∧ x ∈_u y)⌝
      THEN1 (REPEAT_N 2 (once_rewrite_tac [ext_thm])
            THEN prove_tac[get_spec ⌜$<_un⌝]));
a (asm_rewrite_tac[]);
val wf_nat_u_thm = save_pop_thm "wf_nat_u_thm";
```

This allows us to do well-founded induction over the natural number which the way I have implemented it is "course-of-values" induction. However, for the sake of form I will prove that induction principle as an explicit theorem. This is just what you get by expanding the definition of well-foundedness in the above theorem.

SML

```
val nat_u_induct_thm = save_thm ("nat_u_induct_thm",
      (rewrite_rule [get_spec ⌜well_founded⌝] wf_nat_u_thm));
```

Note that this theorem can only be used to prove properties which are true of all sets, so you have to make it conditional (*natural_number*$_u$ *n* ⇒ *whatever*)I suppose I'd better do another one.

SML

```
set_goal([], ⌜∀ p• (∀ x• natural_number_u x ∧ (∀ y• y <_un x ⇒ p y) ⇒ p x)
      ⇒ (∀ x• natural_number_u x ⇒ p x)⌝);
a (asm_tac (rewrite_rule []
      (all_∀_intro (∀_elim ⌜λx• natural_number_u x ⇒ p x⌝ nat_u_induct_thm))));
a (rewrite_tac [all_∀_intro (taut_rule ⌜a ∧ b ⇒ c ⇔ b ⇒ a ⇒ c⌝)]);
a (lemma_tac ⌜∀ p x• (∀ y• y <_un x ⇒ p y)
      ⇔ (∀ y• y <_un x ⇒ natural_number_u y ⇒ p y)⌝);
(* *** Goal "1" *** *)
a (rewrite_tac [get_spec ⌜$<_un⌝]);
a (REPEAT strip_tac THEN all_asm_fc_tac[]);
(* *** Goal "2" *** *)
a (asm_rewrite_tac[]);
val nat_u_induct_thm2 = save_pop_thm "nat_u_induct_thm2";
```

I've tried using that principle and it too has disadvantages. Because $<_{un}$ is used the induction hypothesis is more awkward to use (weaker) than it would have been if $\in_u$ had been used. Unfortunately the proof of an induction theorem using plain set membership is not entirely trivial, so its proof has to be left til later.

4.1 Theorem 2.8

The set of natural numbers.

4.1.1 Natural Numbers are Ordinals

SML

```
set_goal ([], ⌜∀n• natural_number_u n ⇒ Ordinal_u n⌝);
a (rewrite_tac [get_spec ⌜natural_number_u⌝, get_spec ⌜Successor_u⌝]);
a (REPEAT strip_tac THEN_TRY asm_rewrite_tac[ord_u_∅_u_thm]);
a (all_fc_tac [ord_u_suc_u_ord_u_thm]);
val ord_u_nat_u_thm = save_pop_thm "ord_u_nat_u_thm";
```

4.1.2 Members of Natural Numbers are Ordinals

SML

```
set_goal ([], ⌜∀n• natural_number_u n ⇒ ∀m• m ∈_u n ⇒ Ordinal_u m⌝);
a (REPEAT strip_tac);
a (REPEAT (all_fc_tac[ord_u_nat_u_thm, ord_u_∈_u_ord_u_thm]));
val ∈_u_nat_u_ord_u_thm = save_pop_thm "∈_u_nat_u_ord_u_thm";
```

4.1.3 A Natural Number is not a Limit Ordinal

SML

```
set_goal ([], ⌜∀n• natural_number_u n ⇒ ¬ LimitOrdinal_u n⌝);
a (rewrite_tac [get_spec ⌜LimitOrdinal_u⌝, get_spec ⌜natural_number_u⌝]);
a (REPEAT strip_tac);
val nat_u_not_lim_thm = save_pop_thm "nat_u_not_lim_thm";
```

4.1.4 A Natural Number is zero or a Successor

SML

```
set_goal ([], ⌜∀n• natural_number_u n ⇒ Successor_u n ∨ n = ∅_u⌝);
a (rewrite_tac [get_spec ⌜natural_number_u⌝]);
a (REPEAT strip_tac);
val nat_u_zero_or_suc_u_thm = save_pop_thm "nat_u_zero_or_suc_u_thm";
```

4.1.5 A Natural Number does not contain a Limit Ordinal

SML

```
set_goal ([], ⌜∀m n• natural_number_u n ∧ m ∈_u n ⇒ ¬ LimitOrdinal_u m⌝);
a (rewrite_tac [get_spec ⌜LimitOrdinal_u⌝, get_spec ⌜natural_number_u⌝]);
a (REPEAT strip_tac);
(* *** Goal "1" *** *)
a (all_fc_tac [∈_u_not_empty_thm]);
(* *** Goal "2" *** *)
a (all_asm_fc_tac[]);
val ∈_u_nat_u_not_lim_thm = save_pop_thm "∈_u_nat_u_not_lim_thm";
```

4.1.6 All Members of Natural Numbers are Natural Numbers

SML

```
set_goal ([], ⌜∀n• natural_number_u n ⇒ ∀m• m ∈_u n ⇒ natural_number_u m⌝);
a (rewrite_tac [get_spec ⌜natural_number_u⌝]);
a (REPEAT strip_tac THEN_TRY all_asm_fc_tac [∈_u_not_empty_thm]);
a (lemma_tac ⌜Transitive_u n⌝ THEN1
        (REPEAT (all_fc_tac [get_spec ⌜Ordinal_u⌝, Successor_u_ord_u_thm])));
a (lemma_tac ⌜t ∈_u n⌝ THEN1 (EVERY [all_fc_tac [Transitive_u_def],
        POP_ASM_T ante_tac, rewrite_tac [gsu_relext_clauses], asm_prove_tac[]]));
a (all_asm_fc_tac[]);
val ∈_u_nat_u_nat_u_thm = save_pop_thm "∈_u_nat_u_nat_u_thm";
```

```
nat_u_in_G∅_u_thm =
        ⊢ ∀n• natural_number_u n ⇒ n ∈_u Gx_u ∅_u
```

4.1.7 The Existence of w

This comes from the axiom of infinity, however, in galactic set theory we get that from the existence of galaxies, so the following proof is a little unusual.

SML

```
set_goal ([], ⌜∃w• ∀z• z ∈_u w ⇔ natural_number_u z⌝);
a (∃_tac ⌜Sep_u (Gx_u ∅_u) natural_number_u⌝
        THEN rewrite_tac [gsu_opext_clauses]);
a (rewrite_tac [all_∀_intro (taut_rule ⌜(a ∧ b ⇔ b) ⇔ b ⇒ a⌝)]);
a strip_tac;
a (gen_induction_tac1 nat_u_induct_thm2);
a (fc_tac [nat_u_zero_or_suc_u_thm]);
(* *** Goal "1" *** *)
a (fc_tac [get_spec ⌜Successor_u⌝, nat_u_in_G∅_u_thm]);
(* *** Goal "2" *** *)
a (asm_rewrite_tac []);
val ω_u_exists_thm = save_pop_thm "ω_u_exists_thm";
```

4.2 Naming the Natural Numbers

It will be useful to be able to have names for the finite ordinals, which are used as tags in the syntax:

HOL Constant

$\boldsymbol{Nat_u}: \mathbb{N} \rightarrow {}'a\ GSU$

$Nat_u\ 0 = \emptyset_u$
$\land\ \forall n \bullet\ Nat_u\ (n+1) = Suc_u\ (Nat_u\ n)$

We will need to know that these are all distinct ordinals.

$\boldsymbol{Set_u\_Nat_u\_lemma}$ =
$\vdash \forall\ n \bullet\ Set_u\ (Nat_u\ n)$

$\boldsymbol{ord_u\_nat_u\_thm2}$ =
$\vdash \forall\ n \bullet\ Ordinal_u\ (Nat_u\ n)$

$\boldsymbol{not\_suc_u\_nat_u\_zero\_thm}$ =
$\vdash \forall\ n \bullet\ \neg\ Suc_u\ (Nat_u\ n) = \emptyset_u$

$\boldsymbol{lt_u\_sum\_thm}$ =
$\vdash \forall\ x\ y \bullet\ x \leqslant y \Rightarrow (\exists\ z \bullet\ x + z = y)$

$\boldsymbol{nat_u\_mono\_thm}$ =
$\vdash \forall\ x\ y \bullet\ Nat_u\ x \leqslant_u Nat_u\ (x + y)$

$\boldsymbol{nat_u\_one\_one\_thm}$ =
$\vdash \forall\ x\ y \bullet\ Nat_u\ x = Nat_u\ y \Rightarrow x = y$

$\boldsymbol{nat_u\_one\_one\_thm2}$ =
$\vdash \forall\ x\ y \bullet\ Nat_u\ x = Nat_u\ y \Leftrightarrow x = y$

4.3 Proof Context

In this section we define a proof context for natural numbers.

SML

```
add_pc_thms "'gsu-nat" ([nat_u_one_one_thm2]);
add_rw_thms [Set_u_Nat_u_lemma] "'gsu-nat";
add_sc_thms [Set_u_Nat_u_lemma] "'gsu-nat";

set_merge_pcs ["basic_hol", "'gsu-ax", "'gsu-ord", "'gsu-nat"];
commit_pc "'gsu-nat";
```

5 Sequences

SML

```
open_theory "gsu−ord";
force_new_theory "gsu−seq";
new_parent "gsu−fun";
force_new_pc "′gsu−seq";
merge_pcs ["′savedthm_cs_∃_proof"] "′gsu−seq";
set_merge_pcs ["basic_hol", "′gsu−ax", "′gsu−fun", "′gsu−ord", "′gsu−seq"];
```

A sequence is a function whose domain is an ordinal.

HOL Constant

$\boldsymbol{Seq_u}$: $'a\ GSU \to BOOL$

$$\forall s\bullet\ Seq_u\ s \Leftrightarrow Fun_u\ s\ \wedge\ Ordinal_u(Dom_u\ s)$$

The domain is also the length of the sequence.

HOL Constant

$\boldsymbol{Length_u}$: $'a\ GSU \to\ 'a\ GSU$

$$Length_u\ =\ Dom_u$$

5.0.1 Operations on Sequences

A few operations over sequences are defined here, without proving any of their properties (which will be done as and when needed).

HOL Constant

$\boldsymbol{Head_u}$: $'a\ GSU \to\ 'a\ GSU$

$$\forall s\bullet\ Head_u\ s\ =\ s\ {}_u\ \varnothing_u$$

HOL Constant

$\boldsymbol{Front_u}$: $'a\ GSU \to\ 'a\ GSU \to\ 'a\ GSU$

$$\forall \alpha\ s\bullet\ Front_u\ \alpha\ s\ =\ \alpha \lhd_u\ s$$

HOL Constant

$\boldsymbol{Back_u}$: $'a\ GSU \to\ 'a\ GSU \to\ 'a\ GSU$

$$\forall \alpha\ s\bullet\ Back_u\ \alpha\ s\ =\ s\ o_u\ ((\lambda_u\ \beta\bullet\ \beta\ +_u\ \alpha)\ (Dom_u\ s))$$

HOL Constant

$\boldsymbol{Tail_u}: 'a\ GSU \to 'a\ GSU$

$\forall s \bullet\ Tail_u\ s = Back_u\ (Suc_u\ \varnothing_u)\ s$

The symbol $@_u$ will be used for concatenation of sequences.

SML

```
declare_infix(300,"@_u");
```

HOL Constant

$\$\boldsymbol{@_u}: 'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall s\ t \bullet\ s\ @_u\ t = s \cup_u (t\ )$

HOL Constant

$\boldsymbol{UnitSeq_u}: 'a\ GSU \to 'a\ GSU$

$\forall e \bullet\ UnitSeq_u\ e = \varnothing_u \mapsto_u e$

HOL Constant

$\boldsymbol{SeqCons_u}: 'a\ GSU \to 'a\ GSU \to 'a\ GSU$

$\forall e\ s \bullet\ SeqCons_u\ e\ s = (UnitSeq_u\ e)\ @_u\ s$

HOL Constant

$\boldsymbol{SeqDisp_u}: 'a\ GSU\ LIST \to 'a\ GSU$

$SeqDisp_u\ [] = \varnothing_u$

$\land \quad \forall e\ s \bullet\ SeqDisp_u\ (Cons\ e\ s) = SeqCons_u\ e\ (SeqDisp_u\ s)$

5.0.2 Mapping Functions over Sequences

A function can be mapped over a sequence using $RanMap_u$, so we don't need to define a sequence map operator. We probably will need to know that the result of mapping something over a sequence is another sequence of the same length. We already know that mapping over a relation preserves the domain of the relation, so this is trivial.

$\boldsymbol{Seq_u\_RanMap_u\_thm} = \vdash \forall f\ s \bullet\ Seq_u\ s \Rightarrow Seq_u\ (RanMap_u\ f\ s)$

5.0.3 Proof Context

SML

```
add_pc_thms "'gsu-seq" ([]);
add_rw_thms [] "'gsu-seq";
add_sc_thms [] "'gsu-seq";

set_merge_pcs ["basic_hol", "'gsu-ax", "'gsu-fun", "'gsu-ord", "'gsu-seq"];
commit_pc "'gsu-seq";
```

6 Closing

SML

```
open_theory "gsu-ax";
force_new_theory "GSU";
new_parent "gsu-fun";
new_parent "gsu-ord";
new_parent "gsu-nat";
new_parent "gsu-seq";
force_new_pc "'GSU";
force_new_pc "GSU";

val rewrite_thms = ref ([]:THM list);

merge_pcs ["'gsu-ax", "'gsu-fun"(*, "'gsu-sumprod", "'gsu-fixp", "'gsu-lists"*),"'gsu-ord", "'gsu-r
        "'GSU";
commit_pc "'GSU";
merge_pcs ["rbjmisc", "'GSU"] "GSU";
commit_pc "GSU";
```

7 The Theory gsu-ax

7.1 Parents

wf_recp *wf_relp* *U_orders* *rbjmisc*

7.2 Children

GSU *gsu−ord* *gsu−fun*

7.3 Constants

$\boldsymbol{Urelement}$	$'a \rightarrow 'a\ GSU$
$\boldsymbol{UeVal}$	$'a\ GSU \rightarrow 'a$
$\boldsymbol{Ue}$	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{Set_u}$	$'a\ GSU \rightarrow BOOL$
$\$\in_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\boldsymbol{X_u}$	$'a\ GSU \rightarrow 'a\ GSU\ \mathbb{P}$
$\$=_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\$\in_{\boldsymbol{u}}{}^{+}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\$\lhd\in^{+}{}_{\boldsymbol{u}}$	$'a\ GSU \rightarrow ('a\ GSU \rightarrow 'b) \rightarrow 'a\ GSU \rightarrow 'b$
$\$\subseteq_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\$\subset_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\subseteq_{\boldsymbol{u}}\boldsymbol{\_closed}$	$'a\ GSU \rightarrow BOOL$
$\mathbb{P}_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU$
$\bigcup_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{RelIm_u}$	$('a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL) \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Sep_u}$	$'a\ GSU \rightarrow ('a\ GSU \rightarrow BOOL) \rightarrow 'a\ GSU$
$\boldsymbol{galaxy_u}$	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{Gx_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Transitive_u}$	$'a\ GSU \rightarrow BOOL$
$\emptyset_{\boldsymbol{u}}$	$'a\ GSU$
$\boldsymbol{set_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Imagep_u}$	$('a\ GSU \rightarrow 'a\ GSU) \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Pair_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Unit_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\$\cup_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\bigcap_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU$
$\$\cap_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{TrCl_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{ClIm_u}$	$('a\ GSU \rightarrow 'a\ GSU) \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{ClCo_u}$	$('a\ GSU \rightarrow 'a\ GSU) \rightarrow BOOL$
$\boldsymbol{Limit_u}$	$('a\ GSU \rightarrow 'a\ GSU) \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

7.4 Types

'1 ***GSU***

7.5 Fixity

Right Infix 200:

$=_{\boldsymbol{u}}$

Right Infix 230:

$\subseteq_{\boldsymbol{u}} \quad \in_{\boldsymbol{u}} \quad {\in_{\boldsymbol{u}}}^{+} \quad \subset_{\boldsymbol{u}}$

Right Infix 240:

$\cap_{\boldsymbol{u}} \quad \cup_{\boldsymbol{u}}$

Right Infix 300:

$\lhd\in^{+}{}_{\boldsymbol{u}}$

7.6 Axioms

Urelement_Axiom

$\vdash \exists\ f \bullet\ OneOne\ f$

Set_u_axiom $\vdash \forall\ x\ y \bullet\ x \in_u y \Rightarrow Set_u\ y$

gsu_ext_axiom

$\vdash \forall\ s\ t$
$\quad \bullet\ Set_u\ s \wedge Set_u\ t$
$\qquad \Rightarrow (s = t \Leftrightarrow (\forall\ e \bullet\ e \in_u s \Leftrightarrow e \in_u t))$

gsu_wf_axiom $\vdash UWellFounded\ \$\in_u$

UOntology_axiom

$\vdash \forall\ s$
$\quad \bullet\ \exists\ g$
$\qquad \bullet\ s \in_u g$
$\qquad\quad \wedge (\forall\ t$
$\qquad\quad \bullet\ t \in_u g$
$\qquad\qquad \Rightarrow t \subseteq_u g$
$\qquad\qquad\ \wedge (\exists\ p$
$\qquad\qquad\ \bullet\ (\forall\ v \bullet\ v \in_u p \Leftrightarrow Set_u\ v \wedge v \subseteq_u t)$
$\qquad\qquad\qquad \wedge p \in_u g$
$\qquad\qquad\qquad \wedge Set_u\ p)$
$\qquad\qquad\ \wedge (\exists\ u$
$\qquad\qquad\ \bullet\ (\forall\ v$
$\qquad\qquad\qquad \bullet\ v \in_u u \Leftrightarrow (\exists\ w \bullet\ v \in_u w \wedge w \in_u t))$
$\qquad\qquad\qquad \wedge u \in_u g$
$\qquad\qquad\qquad \wedge Set_u\ u)$
$\qquad\qquad\ \wedge (\forall\ rl$
$\qquad\qquad\ \bullet\ ManyOne\ rl$
$\qquad\qquad\qquad \Rightarrow (\exists\ r$
$\qquad\qquad\qquad \bullet\ (\forall\ v$
$\qquad\qquad\qquad\quad \bullet\ v \in_u r$
$\qquad\qquad\qquad\qquad \Leftrightarrow (\exists\ w \bullet\ w \in_u t \wedge rl\ w\ v))$
$\qquad\qquad\qquad\quad \wedge (r \subseteq_u g \Rightarrow r \in_u g)$
$\qquad\qquad\qquad\quad \wedge Set_u\ r)))$

7.7 Definitions

Urelement $\vdash OneOne\ Urelement$

UeVal $\vdash \forall\ x \bullet\ UeVal\ x = (\epsilon\ y \bullet\ Urelement\ y = x)$

Ue $\vdash \forall\ x \bullet\ Ue\ x \Leftrightarrow (\exists\ y \bullet\ x = Urelement\ y)$

$\boldsymbol{Set_u}$ $\quad \vdash \forall\ x \bullet\ Set_u\ x \Leftrightarrow \neg\ (\exists\ y \bullet\ x = Urelement\ y)$

$\boldsymbol{X_u}$ $\quad \vdash \forall\ s \bullet\ X_u\ s = \{t | t \in_u s\}$

$\boldsymbol{=_u}$ $\quad \vdash \forall\ s\ t \bullet\ s =_u t \Leftrightarrow X_u\ s = X_u\ t$

$\boldsymbol{\in_u{}^+}$ $\quad \vdash \$\in_u{}^+ = tc\ \$\in_u$

$\boldsymbol{\lhd\in^+{}_u}$ $\quad \vdash \forall\ s\ f$

$\bullet\ s \lhd\in^+{}_u\ f$

$= (\lambda\ x \bullet\ if\ x \in_u{}^+\ s\ then\ f\ x\ else\ \epsilon\ y \bullet\ T)$

$\boldsymbol{\subseteq_u}$ $\quad \vdash \forall\ s\ t \bullet\ s \subseteq_u t \Leftrightarrow (\forall\ e \bullet\ e \in_u s \Rightarrow e \in_u t)$

$\boldsymbol{\subset_u}$ $\quad \vdash \forall\ s\ t \bullet\ s \subset_u t \Leftrightarrow s \subseteq_u t \land \neg\ t \subseteq_u s$

$\boldsymbol{\subseteq_u\text{-}closed}$ $\quad \vdash \forall\ s$

$\bullet\ \subseteq_u\text{-}closed\ s \Leftrightarrow (\forall\ e\ f \bullet\ e \in_u s \land f \subseteq_u e \Rightarrow f \in_u s)$

$\boldsymbol{\mathbb{P}_u}$ $\quad \vdash \forall\ s$

$\bullet\ Set_u\ (\mathbb{P}_u\ s)$

$\land\ (\forall\ t \bullet\ t \in_u \mathbb{P}_u\ s \Leftrightarrow Set_u\ t \land t \subseteq_u s)$

$\boldsymbol{\bigcup_u}$ $\quad \vdash \forall\ s$

$\bullet\ Set_u\ (\bigcup_u\ s)$

$\land\ (\forall\ t \bullet\ t \in_u \bigcup_u\ s \Leftrightarrow (\exists\ e \bullet\ t \in_u e \land e \in_u s))$

$\boldsymbol{RelIm_u}$ $\quad \vdash \forall\ rl\ s$

$\bullet\ Set_u\ (RelIm_u\ rl\ s)$

$\land\ (ManyOne\ rl$

$\Rightarrow (\forall\ t$

$\bullet\ t \in_u RelIm_u\ rl\ s$

$\Leftrightarrow (\exists\ e \bullet\ e \in_u s \land rl\ e\ t)))$

$\boldsymbol{Sep_u}$ $\quad \vdash \forall\ s\ p$

$\bullet\ (\forall\ e \bullet\ e \in_u Sep_u\ s\ p \Leftrightarrow e \in_u s \land p\ e)$

$\land\ Set_u\ (Sep_u\ s\ p)$

$\boldsymbol{galaxy_u}$ $\quad \vdash \forall\ s$

$\bullet\ galaxy_u\ s$

$\Leftrightarrow (\exists\ x \bullet\ x \in_u s)$

$\land\ (\forall\ t$

$\bullet\ t \in_u s$

$\Rightarrow t \subseteq_u s$

$\land\ \mathbb{P}_u\ t \in_u s$

$\land\ \bigcup_u\ t \in_u s$

$\land\ (\forall\ rl$

$\bullet\ ManyOne\ rl$

$\Rightarrow RelIm_u\ rl\ t \subseteq_u s$

$\Rightarrow RelIm_u\ rl\ t \in_u s))$

$\boldsymbol{Gx_u}$ $\quad \vdash \forall\ s\ t$

$\bullet\ t \in_u Gx_u\ s$

$\Leftrightarrow (\forall\ g \bullet\ galaxy_u\ g \land s \in_u g \Rightarrow t \in_u g)$

$\boldsymbol{Transitive_u}$ $\quad \vdash \forall\ s \bullet\ Transitive_u\ s \Leftrightarrow (\forall\ e \bullet\ e \in_u s \Rightarrow e \subseteq_u s)$

$\boldsymbol{\varnothing_u}$ $\quad \vdash Set_u\ \varnothing_u \land (\forall\ s \bullet\ \neg\ s \in_u \varnothing_u)$

$\boldsymbol{set_u}$ $\quad \vdash \forall\ x \bullet\ set_u\ x = (if\ Set_u\ x\ then\ x\ else\ \varnothing_u)$

$\boldsymbol{Imagep_u}$ $\quad \vdash \forall\ f\ s$

$\bullet\ Set_u\ (Imagep_u\ f\ s)$

$\land\ (\forall\ x$

$\bullet\ x \in_u Imagep_u\ f\ s \Leftrightarrow (\exists\ e \bullet\ e \in_u s \land x = f\ e))$

$\boldsymbol{Pair_u}$ $\quad \vdash \forall\ s\ t$

$\bullet\ Set_u\ (Pair_u\ s\ t)$

$\land\ (\forall\ e \bullet\ e \in_u Pair_u\ s\ t \Leftrightarrow e = s \lor e = t)$

$\boldsymbol{Unit_u}$	$\vdash \forall\ s\bullet\ Unit_u\ s = Pair_u\ s\ s$
$\boldsymbol{\cup_u}$	$\vdash \forall\ s\ t$ $\bullet\ Set_u\ (s \cup_u t)$ $\quad \wedge\ (\forall\ e\bullet\ e \in_u s \cup_u t \Leftrightarrow e \in_u s \vee e \in_u t)$
$\boldsymbol{\bigcap_u}$	$\vdash \forall\ s$ $\bullet\ \bigcap_u s = Sep_u\ (\bigcup_u s)\ (\lambda\ x\bullet\ \forall\ t\bullet\ t \in_u s \Rightarrow x \in_u t)$
$\boldsymbol{\cap_u}$	$\vdash \forall\ s\ t\bullet\ s \cap_u t = Sep_u\ s\ (\lambda\ x\bullet\ x \in_u t)$
$\boldsymbol{TrCl_u}$	$\vdash \forall\ s\bullet\ TrCl_u\ s = Sep_u\ (Gx_u\ s)\ (\lambda\ x\bullet\ x \in_u{}^+ s)$
$\boldsymbol{ClIm_u}$	$\vdash \forall\ f\ \alpha\bullet\ ClIm_u\ f\ \alpha = TrCl_u\ (Imagep_u\ f\ \alpha)$
$\boldsymbol{ClCo_u}$	$\vdash \forall\ f\bullet\ ClCo_u\ f \Leftrightarrow (\forall\ x\ y\bullet\ x \in_u{}^+ y \Rightarrow f\ x \in_u{}^+ f\ y)$
$\boldsymbol{Limit_u}$	$\vdash \forall\ f\ \alpha\bullet\ Limit_u\ f\ \alpha = \bigcup_u (Imagep_u\ f\ \alpha)$

7.8 Theorems

***UeSet_u_lemma*1**
$\vdash \forall\ x\bullet\ Ue\ x \Leftrightarrow \neg\ Set_u\ x$

UWellFounded_well_founded_thm
$\vdash \forall\ \$<<\bullet\ UWellFounded\ \$<< \Leftrightarrow well\_founded\ \$<<$

gsu_ext_thm $\vdash \forall\ s\ t$
$\bullet\ Set_u\ s$
$\quad \Rightarrow Set_u\ t$
$\quad \Rightarrow (s = t \Leftrightarrow (\forall\ e\bullet\ e \in_u s \Leftrightarrow e \in_u t))$

X_u_thm $\vdash \forall\ s\ t\bullet\ s \in X_u\ t \Leftrightarrow s \in_u t$

eq_u_refl_thm
$\vdash \forall\ s\bullet\ s =_u s$

eq_u_sym_thm $\vdash \forall\ s\ t\bullet\ s =_u t \Rightarrow t =_u s$

eq_u_trans_thm
$\vdash \forall\ s\ t\ u\bullet\ s =_u t \wedge t =_u u \Rightarrow t =_u u$

eq_u_ext_thm $\vdash \forall\ s\ t\bullet\ s =_u t \Leftrightarrow (\forall\ u\bullet\ u \in_u s \Leftrightarrow u \in_u t)$

¬eq_u_¬eq_thm
$\vdash \forall\ s\ t\bullet\ \neg\ s =_u t \Rightarrow \neg\ s = t$

***gsu_wf_thm*1** $\vdash well\_founded\ \$\in_u$

gsu_wftc_thm $\vdash well\_founded\ (tc\ \$\in_u)$

gsu_wf_min_thm
$\vdash \forall\ x$
$\bullet\ (\exists\ y\bullet\ y \in_u x)$
$\quad \Rightarrow (\exists\ z\bullet\ z \in_u x \wedge \neg\ (\exists\ v\bullet\ v \in_u z \wedge v \in_u x))$

***gsu_wftc_thm*2**
$\vdash well\_founded\ \$\in_u{}^+$

tc∈_u_incr_thm
$\vdash \forall\ x\ y\bullet\ x \in_u y \Rightarrow x \in_u{}^+ y$

tc∈_u_cases_thm
$\vdash \forall\ x\ y$
$\bullet\ x \in_u{}^+ y \Leftrightarrow x \in_u y \vee (\exists\ z\bullet\ x \in_u{}^+ z \wedge z \in_u y)$

tc∈_u_trans_thm
$\vdash \forall\ s\ t\ u\bullet\ s \in_u{}^+ t \wedge t \in_u{}^+ u \Rightarrow s \in_u{}^+ u$

tc∈_u_decomp_thm
$\vdash \forall\ x\ y$
$\bullet\ x \in_u{}^+ y \wedge \neg\ x \in_u y \Rightarrow (\exists\ z\bullet\ x \in_u{}^+ z \wedge z \in_u y)$

***tc∈_u_decomp_thm*5**
$\vdash \forall\ x\ y$

$\quad \bullet\ x \in_u{}^+ y \wedge \neg\ x \in_u y \Rightarrow (\exists\ z \bullet\ x \in_u z \wedge z \in_u{}^+ y)$

gsu_wf_ind_thm

$\vdash \forall\ p \bullet\ (\forall\ x \bullet\ (\forall\ y \bullet\ y \in_u x \Rightarrow p\ y) \Rightarrow p\ x) \Rightarrow (\forall\ x \bullet\ p\ x)$

gsu_cv_ind_thm

$\vdash \forall\ p$

$\quad \bullet\ (\forall\ x \bullet\ (\forall\ y \bullet\ tc\ \$\in_u\ y\ x \Rightarrow p\ y) \Rightarrow p\ x) \Rightarrow (\forall\ x \bullet\ p\ x)$

gsu_cv_ind_thm2

$\vdash \forall\ p \bullet\ (\forall\ x \bullet\ (\forall\ y \bullet\ y \in_u{}^+ x \Rightarrow p\ y) \Rightarrow p\ x) \Rightarrow (\forall\ x \bullet\ p\ x)$

wf_ul1 $\vdash \forall\ x \bullet\ \neg\ x \in_u x$

wf_ul2 $\vdash \forall\ x\ y \bullet\ \neg\ (x \in_u y \wedge y \in_u x)$

wf_ul3 $\vdash \forall\ x\ y\ z \bullet\ \neg\ (x \in_u y \wedge y \in_u z \wedge z \in_u x)$

$\in_u{}^p$***_recursion_lemma***

$\vdash \forall\ af$

$\quad \bullet\ \exists\ f$

$\quad\quad \bullet\ \forall\ s$

$\quad\quad\quad \bullet\ f\ s$

$\quad\quad\quad\quad = af$

$\quad\quad\quad\quad ((\lambda\ f\ x \bullet\ if\ x \in_u{}^+ s\ then\ f\ x\ else\ \epsilon\ y \bullet\ T)$

$\quad\quad\quad\quad\quad f)$

$\quad\quad\quad\quad s$

$\subseteq_u$***_eq_thm*** $\vdash \forall\ A\ B$

$\quad \bullet\ Set_u\ A \wedge Set_u\ B \Rightarrow (A = B \Leftrightarrow A \subseteq_u B \wedge B \subseteq_u A)$

$\subseteq_u$***_refl_thm*** $\vdash \forall\ A \bullet\ A \subseteq_u A$

$\in_u\subseteq_u$***_def*** $\vdash \forall\ e\ A\ B \bullet\ e \in_u A \wedge A \subseteq_u B \Rightarrow e \in_u B$

$\subseteq_u$***_trans_thm***

$\vdash \forall\ A\ B\ C \bullet\ A \subseteq_u B \wedge B \subseteq_u C \Rightarrow A \subseteq_u C$

$\subset_u$***_trans_thm***

$\vdash \forall\ A\ B\ C \bullet\ A \subset_u B \wedge B \subset_u C \Rightarrow A \subset_u C$

not_$\subset_u$***_thm*** $\vdash \forall\ x \bullet\ \neg\ x \subset_u x$

$\mathbb{P}_u$***_thm*** $\vdash \forall\ s\ t \bullet\ t \in_u \mathbb{P}_u\ s \Leftrightarrow Set_u\ t \wedge t \subseteq_u s$

s$\in\mathbb{P}_{us}$***_thm*** $\vdash \forall\ s \bullet\ Set_u\ s \Rightarrow s \in_u \mathbb{P}_u\ s$

stc$\in\mathbb{P}_{us}$***_thm*** $\vdash \forall\ s \bullet\ Set_u\ s \Rightarrow s \in_u{}^+ \mathbb{P}_u\ s$

$Set_u\mathbb{P}_u$***_thm*** $\vdash \forall\ s \bullet\ Set_u\ (\mathbb{P}_u\ s)$

eq$\mathbb{P}_u$***_thm*** $\vdash \forall\ s\ t$

$\quad \bullet\ Set_u\ s \wedge Set_u\ t$

$\quad\quad \Rightarrow (s = \mathbb{P}_u\ t$

$\quad\quad\quad \Leftrightarrow (\forall\ x \bullet\ x \in_u s \Leftrightarrow Set_u\ x \wedge x \subseteq_u t))$

$\bigcup_u$***_thm*** $\vdash \forall\ s\ t \bullet\ t \in_u \bigcup_u s \Leftrightarrow (\exists\ e \bullet\ t \in_u e \wedge e \in_u s)$

tc$\in_u$***_***$\bigcup_u$***_thm***

$\vdash \forall\ s\ t \bullet\ t \in_u{}^+ \bigcup_u s \Leftrightarrow (\exists\ e \bullet\ t \in_u{}^+ e \wedge e \in_u s)$

$\in_u\bigcup_u$***_thm*** $\vdash \forall\ s\ t \bullet\ t \in_u s \Rightarrow t \subseteq_u \bigcup_u s$

$\in_u\bigcup_u$***_thm2*** $\vdash \forall\ s\ t \bullet\ t \in_u \bigcup_u s \Rightarrow (\exists\ e \bullet\ t \in_u e \wedge e \in_u s)$

$\in_u\bigcup_u$***_thm3*** $\vdash \forall\ s\ t \bullet\ (\exists\ e \bullet\ t \in_u e \wedge e \in_u s) \Rightarrow t \in_u \bigcup_u s$

$Set_u\bigcup_u$***_thm*** $\vdash \forall\ s \bullet\ Set_u\ (\bigcup_u s)$

$\bigcup_u$***_ext_thm*** $\vdash \forall\ x\ y$

$\quad \bullet\ \bigcup_u x = y$

$\quad\quad \Leftrightarrow Set_u\ y$

$\quad\quad\quad \wedge (\forall\ z \bullet\ z \in_u y \Leftrightarrow (\exists\ w \bullet\ z \in_u w \wedge w \in_u x))$

$Set_{uRelIm u}$***_thm***

$\vdash \forall\ rl\ s \bullet\ Set_u\ (RelIm_u\ rl\ s)$

$RelIm_u$***_thm*** $\vdash \forall\ rl\ s$

$\bullet\ ManyOne\ rl$
$\quad \Rightarrow (\forall\ t$
$\quad \bullet\ t \in_u RelIm_u\ rl\ s \Leftrightarrow (\exists\ e \bullet\ e \in_u s \land rl\ e\ t))$

Sep_u_thm $\vdash \forall\ s\ p\ e \bullet\ e \in_u Sep_u\ s\ p \Leftrightarrow e \in_u s \land p\ e$

Set_u_Sep_u_thm
$\vdash \forall\ s\ p \bullet\ Set_u\ (Sep_u\ s\ p)$

Sep_u_⊆_u_thm
$\vdash \forall\ s\ p \bullet\ Set_u\ s \Rightarrow Sep_u\ s\ p \subseteq_u s$

Sep_u_sub_thm
$\vdash \forall\ s\ p\ e \bullet\ e \in_u Sep_u\ s\ p \Rightarrow e \in_u s$

Sep_u_∈_u_ℙ_u_thm
$\vdash \forall\ s\ p \bullet\ Set_u\ s \Rightarrow Sep_u\ s\ p \in_u \mathbb{P}_u\ s$

Sep_u_⊆_thm $\vdash \forall\ s\ t \bullet\ Set_u\ t \land t \subseteq_u s \Rightarrow Sep_u\ s\ (CombC\ \$\in_u\ t) = t$

galaxies_u_∃_thm
$\vdash \forall\ s \bullet\ \exists\ g \bullet\ s \in_u g \land galaxy_u\ g$

galaxy_u_Set_u_thm
$\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow Set_u\ g$

t_in_Gx_u_t_thm
$\vdash \forall\ t \bullet\ t \in_u Gx_u\ t$

tc∈_u_Gx_u_thm
$\vdash \forall\ t \bullet\ t \in_u{}^+ Gx_u\ t$

Set_u_Gx_u_thm
$\vdash \forall\ x \bullet\ Set_u\ (Gx_u\ x)$

Gx_u_⊆_u_galaxy_u
$\vdash \forall\ s\ g \bullet\ galaxy_u\ g \land s \in_u g \Rightarrow Gx_u\ s \subseteq_u g$

galaxy_u_Gx_u
$\vdash \forall\ s \bullet\ galaxy_u\ (Gx_u\ s)$

galaxies_u_transitive_thm
$\vdash \forall\ s \bullet\ galaxy_u\ s \Rightarrow Transitive_u\ s$

GClose_uSep_u_thm
$\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\quad \Rightarrow (\forall\ s \bullet\ s \in_u g \Rightarrow (\forall\ p \bullet\ Sep_u\ s\ p \in_u g))$

GClose_u_fc_clauses
$\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\quad \Rightarrow (\forall\ s$
$\quad \bullet\ s \in_u g$
$\qquad \Rightarrow \mathbb{P}_u\ s \in_u g$
$\qquad \land \bigcup_u s \in_u g$
$\qquad \land (\forall\ p \bullet\ Sep_u\ s\ p \in_u g)$
$\qquad \land (\forall\ t \bullet\ Set_u\ t \land t \subseteq_u s \Rightarrow t \in_u g))$

GClose_u_tc∈_u_thm
$\vdash \forall\ s\ g \bullet\ galaxy_u\ g \Rightarrow s \in_u{}^+ g \Rightarrow s \in_u g$

GClose_u_tc∈_u_thm2
$\vdash \forall\ t\ s\ g \bullet\ galaxy_u\ g \land t \in_u g \land s \in_u{}^+ t \Rightarrow s \in_u g$

Gx_u_mono_thm
$\vdash \forall\ s\ t$
$\bullet\ Set_u\ s \land Set_u\ t \land s \subseteq_u t \Rightarrow Gx_u\ s \subseteq_u Gx_u\ t$

Gx_u_mono_thm2
$\vdash \forall\ s\ t \bullet\ s \in_u t \Rightarrow Gx_u\ s \subseteq_u Gx_u\ t$

$\boldsymbol{Gx_u\_trans\_thm}$

$\vdash \forall\ s \bullet\ Transitive_u\ (Gx_u\ s)$

$\boldsymbol{t\_\subseteq_u\_Gx_u\_t\_thm}$

$\vdash \forall\ t \bullet\ t \subseteq_u\ Gx_u\ t$

$\boldsymbol{Gx_u\_mono\_thm3}$

$\vdash \forall\ s\ t \bullet\ s \subseteq_u\ t \Rightarrow s \subseteq_u\ Gx_u\ t$

$\boldsymbol{Gx_u\_mono\_thm4}$

$\vdash \forall\ s\ t \bullet\ Set_u\ s \land s \subseteq_u\ t \Rightarrow s \in_u\ Gx_u\ t$

$\boldsymbol{Gx_u\_trans\_thm2}$

$\vdash \forall\ s\ t \bullet\ s \in_u\ t \Rightarrow s \in_u\ Gx_u\ t$

$\boldsymbol{Gx_u\_trans\_thm3}$

$\vdash \forall\ s\ t\ u \bullet\ s \in_u\ t \land t \in_u\ Gx_u\ u \Rightarrow s \in_u\ Gx_u\ u$

$\boldsymbol{Gx_u\_trans\_thm4}$

$\vdash \forall\ s\ t\ u \bullet\ s \in_u{}^+\ t \land t \in_u\ Gx_u\ u \Rightarrow s \in_u\ Gx_u\ u$

$\boldsymbol{Gx_u\_trans\_thm5}$

$\vdash \forall\ s\ t\ u \bullet\ s \in_u{}^+\ t \Rightarrow s \in_u\ Gx_u\ t$

$\boldsymbol{tc\in_u\_\emptyset_u\_thm}$

$\vdash \forall\ x \bullet\ \neg\ x \in_u{}^+\ \emptyset_u$

$\boldsymbol{X\emptyset_u\_thm}$ $\vdash \forall\ x \bullet\ \neg\ x \in X_u\ \emptyset_u$

$\boldsymbol{eq\_\emptyset_u\_\neg\in_u\_thm}$

$\vdash \forall\ x \bullet\ x =_u\ \emptyset_u \Rightarrow (\forall\ y \bullet\ \neg\ y \in_u\ x)$

$\boldsymbol{eq\_\emptyset_u\_\neg tc\in_u\_thm}$

$\vdash \forall\ x \bullet\ x =_u\ \emptyset_u \Rightarrow (\forall\ y \bullet\ \neg\ y \in_u{}^+\ x)$

$\boldsymbol{eq_u\_eq\_\emptyset_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Set_u\ \alpha \land \alpha =_u\ \emptyset_u \Rightarrow \alpha = \emptyset_u$

$\boldsymbol{G\emptyset_{uC}}$ $\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow \emptyset_u \in_u\ g$

$\boldsymbol{\emptyset_u\subseteq_u\_thm}$ $\vdash \forall\ s \bullet\ \emptyset_u \subseteq_u\ s$

$\boldsymbol{\in_u\_not\_empty\_thm}$

$\vdash \forall\ m\ n \bullet\ m \in_u\ n \Rightarrow \neg\ n = \emptyset_u$

$\boldsymbol{\emptyset_u\_\in_u\_galaxy_u\_thm}$

$\vdash \forall\ x \bullet\ galaxy_u\ x \Rightarrow \emptyset_u \in_u\ x$

$\boldsymbol{\emptyset_u\_\in_u\_Gx_u\_thm}$

$\vdash \forall\ x \bullet\ \emptyset_u \in_u\ Gx_u\ x$

$\boldsymbol{Set_u\_set_u\_thm}$

$\vdash \forall\ s \bullet\ Set_u\ (set_u\ s)$

$\boldsymbol{set_u\_eq_u\_thm}$

$\vdash \forall\ s \bullet\ set_u\ s =_u\ s$

$\boldsymbol{set_u\_eq_u\_thm2}$

$\vdash \forall\ s\ u \bullet\ u \in_u\ set_u\ s \Leftrightarrow u \in_u\ s$

$\boldsymbol{set_u\_fc\_thm}$ $\vdash \forall\ s \bullet\ Set_u\ s \Rightarrow set_u\ s = s$

$\boldsymbol{Imagep_u\_\emptyset_u\_thm}$

$\vdash \forall\ f \bullet\ Imagep_u\ f\ \emptyset_u = \emptyset_u$

$\boldsymbol{tc\in_u\_Imagep_u\_thm}$

$\vdash \forall\ f\ s\ x$

$\bullet\ x \in_u{}^+\ Imagep_u\ f\ s$

$\Leftrightarrow (\exists\ y \bullet\ y \in_u\ s \land (x = f\ y \lor x \in_u{}^+\ f\ y))$

$\boldsymbol{Imagep_u\_comp\_thm}$

$\vdash \forall\ s\ f\ g$

$\bullet\ Imagep_u\ f\ (Imagep_u\ g\ s) = Imagep_u\ (f\ o\ g)\ s$

$\boldsymbol{GImagep_{uC}}$ $\vdash \forall\ g$

$\bullet\ galaxy_u\ g$

$\Rightarrow (\forall\ s$
$\bullet\ s \in_u g$
$\Rightarrow (\forall\ f$
$\bullet\ Imagep_u\ f\ s \subseteq_u g \Rightarrow Imagep_u\ f\ s \in_u g))$

Pair$_u$_$\in_u$_thm

$\vdash \forall\ x\ y \bullet\ x \in_u Pair_u\ x\ y \land y \in_u Pair_u\ x\ y$

Pair$_u$_tc$\in_u$_thm

$\vdash \forall\ s\ t \bullet\ s \in_u{}^+ Pair_u\ s\ t \land t \in_u{}^+ Pair_u\ s\ t$

Pair$_u$_eq_thm

$\vdash \forall\ s\ t\ u\ v$
$\bullet\ Pair_u\ s\ t = Pair_u\ u\ v$
$\Leftrightarrow s = u \land t = v \lor s = v \land t = u$

GClose$_{uPairu}$

$\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\Rightarrow (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow Pair_u\ s\ t \in_u g)$

Unit$_u$_thm2 $\vdash \forall\ x \bullet\ x \in_u Unit_u\ x$

Unit$_u$_tc$\in_u$_thm

$\vdash \forall\ x \bullet\ x \in_u{}^+ Unit_u\ x$

GClose$_{uUnitu}$

$\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow (\forall\ s \bullet\ s \in_u g \Rightarrow Unit_u\ s \in_u g)$

$\subseteq_u\cup_u$_thm $\vdash \forall\ A\ B \bullet\ A \subseteq_u A \cup_u B \land B \subseteq_u A \cup_u B$

$\cup_u\subseteq_u$_def1 $\vdash \forall\ A\ B\ C \bullet\ A \subseteq_u C \land B \subseteq_u C \Rightarrow A \cup_u B \subseteq_u C$

$\cup_u\subseteq_u$_def2 $\vdash \forall\ A\ B\ C\ D \bullet\ A \subseteq_u C \land B \subseteq_u D \Rightarrow A \cup_u B \subseteq_u C \cup_u D$

$\cup_u\emptyset_u$_clauses

$\vdash \forall\ A \bullet\ Set_u\ A \Rightarrow A \cup_u \emptyset_u = A \land \emptyset_u \cup_u A = A$

$\cup_u$_comm_thm $\vdash \forall\ A\ B \bullet\ A \cup_u B = B \cup_u A$

$\cup_u$_$\emptyset_u$_set$_u$_thm

$\vdash \forall\ A \bullet\ A \cup_u \emptyset_u = set_u\ A \land \emptyset_u \cup_u A = set_u\ A$

tc$\in_u$_$\cup_u$_thm

$\vdash \forall\ x\ A\ B \bullet\ x \in_u{}^+ A \cup_u B \Leftrightarrow x \in_u{}^+ A \lor x \in_u{}^+ B$

GClose$_u\cup_u$ $\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\Rightarrow (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow s \cup_u t \in_u g)$

$\bigcap_u$_thm $\vdash \forall\ x\ s\ e$
$\bullet\ x \in_u s \Rightarrow (e \in_u \bigcap_u s \Leftrightarrow (\forall\ y \bullet\ y \in_u s \Rightarrow e \in_u y))$

$\bigcap_u\subseteq_u$_thm $\vdash \forall\ s\ t \bullet\ s \in_u t \Rightarrow \bigcap_u t \subseteq_u s$

$\subseteq_u\bigcap_u$_thm $\vdash \forall\ A\ B$
$\bullet\ A \in_u B$
$\Rightarrow (\forall\ C \bullet\ (\forall\ D \bullet\ D \in_u B \Rightarrow C \subseteq_u D) \Rightarrow C \subseteq_u \bigcap_u B)$

$\bigcap_u\emptyset_u$_thm $\vdash \bigcap_u \emptyset_u = \emptyset_u$

GClose$_u\bigcap_u$ $\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow (\forall\ s \bullet\ s \in_u g \Rightarrow \bigcap_u s \in_u g)$

GClose$_u\cap_u$ $\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\Rightarrow (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow s \cap_u t \in_u g)$

$\cap_u$_thm $\vdash \forall\ s\ t\ e \bullet\ e \in_u s \cap_u t \Leftrightarrow e \in_u s \land e \in_u t$

$\subseteq_u\cap_u$_thm $\vdash \forall\ A\ B \bullet\ A \cap_u B \subseteq_u A \land A \cap_u B \subseteq_u B$

$\cap_u\subseteq_u$_def1 $\vdash \forall\ A\ B\ C \bullet\ A \subseteq_u C \land B \subseteq_u C \Rightarrow A \cap_u B \subseteq_u C$

$\cap_u\subseteq_u$_def2 $\vdash \forall\ A\ B\ C\ D \bullet\ A \subseteq_u C \land B \subseteq_u D \Rightarrow A \cap_u B \subseteq_u C \cap_u D$

$\cap_u\subseteq_u$_def3 $\vdash \forall\ A\ B\ C \bullet\ C \subseteq_u A \land C \subseteq_u B \Rightarrow C \subseteq_u A \cap_u B$

not_x_$\in_u$_x_thm

$\vdash \neg (\exists\ x \bullet\ x \in_u\ x)$

GClose$_u$_fc_clauses2

$\vdash \forall\ g$
$\bullet\ galaxy_u\ g$
$\Rightarrow (\forall\ s\ t \bullet\ s \in_u\ g \land t \in_u\ g \Rightarrow Pair_u\ s\ t \in_u\ g)$
$\land (\forall\ s \bullet\ s \in_u\ g \Rightarrow Unit_u\ s \in_u\ g)$
$\land (\forall\ s\ t \bullet\ s \in_u\ g \land t \in_u\ g \Rightarrow s \cup_u\ t \in_u\ g)$
$\land (\forall\ s \bullet\ s \in_u\ g \Rightarrow \bigcap_u\ s \in_u\ g)$
$\land (\forall\ s\ t \bullet\ s \in_u\ g \land t \in_u\ g \Rightarrow s \cap_u\ t \in_u\ g)$

tc$\in_u$_clauses

$\vdash \forall\ s$
$\bullet\ s \in_u{}^+\ Unit_u\ s$
$\land (\forall\ t \bullet\ t \in_u{}^+\ Pair_u\ s\ t \land s \in_u{}^+\ Pair_u\ s\ t)$

Set$_u$_TrCl$_u$_thm

$\vdash \forall\ s \bullet\ Set_u\ (TrCl_u\ s)$

TrCl$_u$_sup_thm

$\vdash \forall\ s \bullet\ s \subseteq_u\ TrCl_u\ s$

TrCl$_u$_sup_thm2

$\vdash \forall\ s\ t \bullet\ Transitive_u\ t \land s \subseteq_u\ t \Rightarrow TrCl_u\ s \subseteq_u\ t$

Transitive$_u$_TrCl$_u$_thm

$\vdash \forall\ s \bullet\ Transitive_u\ (TrCl_u\ s)$

TrCl$_u$_ext_thm

$\vdash \forall\ s\ x$
$\bullet\ x \in_u\ TrCl_u\ s$
$\Leftrightarrow (\forall\ t \bullet\ Transitive_u\ t \land s \subseteq_u\ t \Rightarrow x \in_u\ t)$

TrCl$_u$_ext_thm2

$\vdash \forall\ s\ t \bullet\ s \in_u\ TrCl_u\ t \Leftrightarrow s \in_u{}^+\ t$

tc$\in_u$_TrCl$_u$_thm

$\vdash \forall\ s\ t \bullet\ s \in_u{}^+\ TrCl_u\ t \Leftrightarrow s \in_u{}^+\ t$

Tran_set_TrCl_thm

$\vdash \forall\ s \bullet\ Set_u\ s \land Transitive_u\ s \Rightarrow TrCl_u\ s = s$

Tran_set_tc$\in_u$_thm

$\vdash \forall\ s$
$\bullet\ Set_u\ s \land Transitive_u\ s$
$\Rightarrow (\forall\ x \bullet\ x \in_u{}^+\ s \Rightarrow x \in_u\ s)$

Tran_tc$\in_u$_thm

$\vdash \forall\ s \bullet\ Transitive_u\ s \Rightarrow (\forall\ x \bullet\ x \in_u{}^+\ s \Rightarrow x \in_u\ s)$

Set$_u$_ClIm$_u$_thm

$\vdash \forall\ f\ \alpha \bullet\ Set_u\ (ClIm_u\ f\ \alpha)$

$\in_u$_ClIm$_u$_thm

$\vdash \forall\ f\ \alpha\ x$
$\bullet\ x \in_u\ ClIm_u\ f\ \alpha$
$\Leftrightarrow (\exists\ y \bullet\ y \in_u\ \alpha \land (x = f\ y \lor x \in_u{}^+\ f\ y))$

tc$\in_u$_ClIm$_u$_thm

$\vdash \forall\ f\ \alpha\ x$
$\bullet\ x \in_u{}^+\ ClIm_u\ f\ \alpha$
$\Leftrightarrow (\exists\ y \bullet\ y \in_u\ \alpha \land (x = f\ y \lor x \in_u{}^+\ f\ y))$

ClIm$_u$_$\varnothing_u$_thm

$\vdash \forall\ f\ s \bullet\ Set_u\ s \Rightarrow (ClIm_u\ f\ s = \varnothing_u \Leftrightarrow s = \varnothing_u)$

ClIm$_u$_$\varnothing_u$_thm2

$\vdash \forall\ f\ s \bullet\ ClIm_u\ f\ s = \varnothing_u \Leftrightarrow s =_u\ \varnothing_u$

ClIm$_u$_∅$_u$_thm3

$\vdash \forall\ f\bullet\ ClIm_u\ f\ \varnothing_u\ =\ \varnothing_u$

ClIm$_u$_ext_thm

$\vdash \forall\ f\ s\ t\bullet\ s\ =_u\ t\ \Rightarrow\ ClIm_u\ f\ s\ =\ ClIm_u\ f\ t$

ClIm$_u$_mono_thm

$\vdash \forall\ f\ s\ t\bullet\ s\ \subseteq_u\ t\ \Rightarrow\ ClIm_u\ f\ s\ \subseteq_u\ ClIm_u\ f\ t$

ClCo$_u$_ClIm$_u$_thm

$\vdash \forall\ s\ f\ g$
$\bullet\ ClCo_u\ f\ \Rightarrow\ ClIm_u\ f\ (ClIm_u\ g\ s)\ =\ ClIm_u\ (f\ o\ g)\ s$

Set$_u$_Limit$_u$_thm

$\vdash \forall\ f\ \alpha\bullet\ Set_u\ (Limit_u\ f\ \alpha)$

∈$_u$_Limit$_u$_thm

$\vdash \forall\ f\ \alpha\ x$
$\bullet\ x\ \in_u\ Limit_u\ f\ \alpha\ \Leftrightarrow\ (\exists\ y\bullet\ y\ \in_u\ \alpha\ \wedge\ x\ \in_u\ f\ y)$

tc∈$_u$_Limit$_u$_thm

$\vdash \forall\ f\ \alpha\ x$
$\bullet\ x\ {\in_u}^+\ Limit_u\ f\ \alpha\ \Leftrightarrow\ (\exists\ y\bullet\ y\ \in_u\ \alpha\ \wedge\ x\ {\in_u}^+\ f\ y)$

◁tc∈$_u$_respects_∈$_u$_thm

$\vdash \forall\ af\bullet\ (\lambda\ f\ x\bullet\ af\ (x\ \lhd{\in^+}_u\ f)\ x)\ respects\ \$\in_u$

◁tc∈$_u$_recursion_thm

$\vdash \forall\ af\bullet\ \exists\ f\bullet\ \forall\ x\bullet\ f\ (CombI\ x)\ =\ af\ (x\ \lhd{\in^+}_u\ f)\ x$

Imagep$_u$_respects_∈$_u$_thm

$\vdash \forall\ af\bullet\ (\lambda\ f\ x\bullet\ af\ (Imagep_u\ f\ x)\ x)\ respects\ \$\in_u$

Imagep$_u$_recursion_thm

$\vdash \forall\ af\bullet\ \exists\ f\bullet\ \forall\ x\bullet\ f\ (CombI\ x)\ =\ af\ (Imagep_u\ f\ x)\ x$

ClIm$_u$_respects_∈$_u$_thm

$\vdash \forall\ af\bullet\ (\lambda\ f\ x\bullet\ af\ (ClIm_u\ f\ x)\ x)\ respects\ \$\in_u$

ClIm$_u$_recursion_thm

$\vdash \forall\ af\bullet\ \exists\ f\bullet\ \forall\ x\bullet\ f\ (CombI\ x)\ =\ af\ (ClIm_u\ f\ x)\ x$

Set$_u$_clauses

$\vdash \forall\ x\ y\ r\ p\ f$
$\bullet\ Set_u\ (\mathbb{P}_u\ x)$
$\wedge\ Set_u\ (\bigcup_u\ x)$
$\wedge\ Set_u\ (RelIm_u\ r\ x)$
$\wedge\ Set_u\ (Sep_u\ x\ p)$
$\wedge\ Set_u\ (Gx_u\ x)$
$\wedge\ Set_u\ \varnothing_u$
$\wedge\ Set_u\ (Imagep_u\ f\ y)$
$\wedge\ Set_u\ (Pair_u\ x\ y)$
$\wedge\ Set_u\ (Unit_u\ x)$
$\wedge\ Set_u\ (x\ \cup_u\ y)$
$\wedge\ Set_u\ (\bigcap_u\ x)$
$\wedge\ Set_u\ (x\ \cap_u\ y)$
$\wedge\ Set_u\ (TrCl_u\ x)$
$\wedge\ Set_u\ (ClIm_u\ f\ x)$

gsu_opext_clauses

$\vdash \forall\ s\ x\ f\ t\ u\ v\ e\ p$
$\bullet\ \neg\ s\ \in_u\ \varnothing_u$
$\wedge\ (Set_u\ (\mathbb{P}_u\ s)$
$\wedge\ (\forall\ t\bullet\ t\ \in_u\ \mathbb{P}_u\ s\ \Leftrightarrow\ Set_u\ t\ \wedge\ t\ \subseteq_u\ s))$
$\wedge\ (Set_u\ (\bigcup_u\ s)$

$\wedge\ (\forall\ t$
$\bullet\ t \in_u \bigcup_u\ s \Leftrightarrow (\exists\ e \bullet\ t \in_u\ e\ \wedge\ e \in_u\ s)))$
$\wedge\ (x \in_u\ Imagep_u\ f\ s \Leftrightarrow (\exists\ e \bullet\ e \in_u\ s\ \wedge\ x = f\ e))$
$\wedge\ Imagep_u\ f\ \varnothing_u = \varnothing_u$
$\wedge\ (x \in_u{}^+\ Imagep_u\ f\ s$
$\Leftrightarrow (\exists\ y \bullet\ y \in_u\ s\ \wedge\ (x = f\ y\ \vee\ x \in_u{}^+\ f\ y)))$
$\wedge\ (Pair_u\ s\ t = Pair_u\ u\ v$
$\Leftrightarrow s = u\ \wedge\ t = v\ \vee\ s = v\ \wedge\ t = u)$
$\wedge\ (Set_u\ (Pair_u\ s\ t)$
$\wedge\ (\forall\ e \bullet\ e \in_u\ Pair_u\ s\ t \Leftrightarrow e = s\ \vee\ e = t))$
$\wedge\ (Unit_u\ s = Unit_u\ t \Leftrightarrow s = t)$
$\wedge\ (e \in_u\ Unit_u\ s \Leftrightarrow e = s)$
$\wedge\ (Pair_u\ s\ t = Unit_u\ u \Leftrightarrow s = u\ \wedge\ t = u)$
$\wedge\ (Unit_u\ s = Pair_u\ t\ u \Leftrightarrow s = t\ \wedge\ s = u)$
$\wedge\ (e \in_u\ Sep_u\ s\ p \Leftrightarrow e \in_u\ s\ \wedge\ p\ e)$
$\wedge\ (Set_u\ (s \cup_u\ t)$
$\wedge\ (\forall\ e \bullet\ e \in_u\ s \cup_u\ t \Leftrightarrow e \in_u\ s\ \vee\ e \in_u\ t))$
$\wedge\ (e \in_u\ s \cap_u\ t \Leftrightarrow e \in_u\ s\ \wedge\ e \in_u\ t)$

gsu_relext_clauses

$\vdash\ \forall\ s\ t$
$\bullet\ (s \subseteq_u\ t \Leftrightarrow (\forall\ e \bullet\ e \in_u\ s \Rightarrow e \in_u\ t))$
$\wedge\ (s =_u\ t \Leftrightarrow (\forall\ u \bullet\ u \in_u\ s \Leftrightarrow u \in_u\ t))$

8 The Theory gsu-fun

8.1 Parents

gsu−ax

8.2 Children

GSU gsu−seq

8.3 Constants

$\$\mapsto_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\boldsymbol{Snd_u}$	$'a\ GSU \to 'a\ GSU$
$\boldsymbol{Fst_u}$	$'a\ GSU \to 'a\ GSU$
$\boldsymbol{MkPair_u}$	$'a\ GSU \times 'a\ GSU \to 'a\ GSU$
$\boldsymbol{MkTriple_u}$	$'a\ GSU \times 'a\ GSU \times 'a\ GSU \to 'a\ GSU$
$\boldsymbol{Rel_u}$	$'a\ GSU \to BOOL$
$\boldsymbol{Dom_u}$	$'a\ GSU \to 'a\ GSU$
$\boldsymbol{Ran_u}$	$'a\ GSU \to 'a\ GSU$
$\boldsymbol{RanMap_u}$	$('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$
$\boldsymbol{Field_u}$	$'a\ GSU \to 'a\ GSU$
$\$\lhd_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\$\rhd_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\$\ntriangleleft_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\$\ntriangleright_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\boldsymbol{Rel2DepType_u}$	$'a\ GSU \to 'a\ GSU$
$\boldsymbol{DepSum_u}$	$'a\ GSU \to 'a\ GSU$
$\$\boldsymbol{\Sigma_u}$	$('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$
$\$\times_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\$\leftrightarrow_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\$\boldsymbol{o_u}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\boldsymbol{Fun_u}$	$'a\ GSU \to BOOL$
$\$\nrightarrow_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\$\rightarrow_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\rightarrow_{\boldsymbol{u}}\boldsymbol{\_closed}$	$'a\ GSU \to BOOL$
$\$\boldsymbol{\lambda_u}$	$('a\ GSU \to 'a\ GSU) \to 'a\ GSU \to 'a\ GSU$
$\$_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$
$\boldsymbol{Id_u}$	$'a\ GSU \to 'a\ GSU$
$\$\oplus_{\boldsymbol{u}}$	$'a\ GSU \to 'a\ GSU \to 'a\ GSU$

8.4 Fixity

Binder: $\boldsymbol{\Sigma_u}$ $\boldsymbol{\lambda_u}$

Right Infix 240:

$\leftrightarrow_{\boldsymbol{u}}$ $\rightarrow_{\boldsymbol{u}}$ $\times_{\boldsymbol{u}}$ $\nrightarrow_{\boldsymbol{u}}$ $\mapsto_{\boldsymbol{u}}$

Right Infix 250:

$\boldsymbol{o_u}$ $_{\boldsymbol{u}}$ $\oplus_{\boldsymbol{u}}$

Right Infix 300:

$\ntriangleright_{\boldsymbol{u}}$ $\rhd_{\boldsymbol{u}}$ $\ntriangleleft_{\boldsymbol{u}}$ $\lhd_{\boldsymbol{u}}$

8.5 Definitions

$\boldsymbol{\mapsto_u}$ $\vdash \forall\ s\ t\bullet\ s \mapsto_u t = Pair_u\ (Unit_u\ s)\ (Pair_u\ s\ t)$

$\boldsymbol{Fst_u}$

$\boldsymbol{Snd_u}$ $\vdash \forall\ s\ t\bullet\ Fst_u\ (s \mapsto_u t) = s \land Snd_u\ (s \mapsto_u t) = t$

$\boldsymbol{MkPair_u}$ $\vdash \forall\ lr\bullet\ MkPair_u\ lr = Fst\ lr \mapsto_u Snd\ lr$

$\boldsymbol{MkTriple_u}$ $\vdash \forall\ t\bullet\ MkTriple_u\ t = Fst\ t \mapsto_u MkPair_u\ (Snd\ t)$

$\boldsymbol{Rel_u}$ $\vdash \forall\ x\bullet\ Rel_u\ x \Leftrightarrow (\forall\ y\bullet\ y \in_u x \Rightarrow (\exists\ s\ t\bullet\ y = s \mapsto_u t))$

$\boldsymbol{Dom_u}$ $\vdash \forall\ x$
$\bullet\ Dom_u\ x = Sep_u\ (Gx_u\ x)\ (\lambda\ w\bullet\ \exists\ v\bullet\ w \mapsto_u v \in_u x)$

$\boldsymbol{Ran_u}$ $\vdash \forall\ x$
$\bullet\ Ran_u\ x = Sep_u\ (Gx_u\ x)\ (\lambda\ w\bullet\ \exists\ v\bullet\ v \mapsto_u w \in_u x)$

$\boldsymbol{RanMap_u}$ $\vdash \forall\ f\ s$
$\bullet\ RanMap_u\ f\ s$
$= Imagep_u\ (\lambda\ e\bullet\ Fst_u\ e \mapsto_u f\ (Snd_u\ e))\ s$

$\boldsymbol{Field_u}$ $\vdash \forall\ s\bullet\ Field_u\ s = Dom_u\ s \cup_u Ran_u\ s$

$\boldsymbol{\lhd_u}$ $\vdash \forall\ s\ r\bullet\ s \lhd_u r = Sep_u\ r\ (\lambda\ p\bullet\ Fst_u\ p \in_u s)$

$\boldsymbol{\rhd_u}$ $\vdash \forall\ s\ r\bullet\ r \rhd_u s = Sep_u\ r\ (\lambda\ p\bullet\ Snd_u\ p \in_u s)$

$\boldsymbol{⩤_u}$ $\vdash \forall\ s\ r\bullet\ s ⩤_u r = Sep_u\ r\ (\lambda\ p\bullet\ \neg\ Fst_u\ p \in_u s)$

$\boldsymbol{⩥_u}$ $\vdash \forall\ s\ r\bullet\ r ⩥_u s = Sep_u\ r\ (\lambda\ p\bullet\ \neg\ Snd_u\ p \in_u s)$

$\boldsymbol{Rel2DepType_u}$
$\vdash \forall\ r$
$\bullet\ Rel2DepType_u\ r$
$= Sep_u$
$(Gx_u\ r)$
$(\lambda\ e$
$\bullet\ \exists\ i\ t$
$\bullet\ e = i \mapsto_u t$
$\land\ i \in_u Dom_u\ r$
$\land\ (\forall\ j\bullet\ j \in_u t \Leftrightarrow i \mapsto_u j \in_u r))$

$\boldsymbol{DepSum_u}$ $\vdash \forall\ t$
$\bullet\ DepSum_u\ t$
$= Sep_u$
$(Gx_u\ t)$
$(\lambda\ e$
$\bullet\ \exists\ i\ t2\ v$
$\bullet\ e = i \mapsto_u v \land v \in_u t2 \land i \mapsto_u t2 \in_u t)$

$\boldsymbol{\Sigma_u}$ $\vdash \forall\ f\ s$
$\bullet\ \$\Sigma_u\ f\ s$
$= \bigcup_u$
$(Imagep_u$
$(\lambda\ e\bullet\ Imagep_u\ (\lambda\ x\bullet\ e \mapsto_u x)\ (f\ e))$
$s)$

$\boldsymbol{\times_u}$ $\vdash \forall\ s\ t$
$\bullet\ s \times_u t$
$= \bigcup_u$
$(Imagep_u$
$(\lambda\ se\bullet\ Imagep_u\ (\lambda\ te\bullet\ se \mapsto_u te)\ t)$
$s)$

$\boldsymbol{\leftrightarrow_u}$ $\vdash \forall\ s\ t\bullet\ s \leftrightarrow_u t = \mathbb{P}_u\ (s \times_u t)$

$\boldsymbol{o_u}$ $\vdash \forall\ f\ g$
$\bullet\ f\ o_u\ g$

$= Imagep_u$
$(\lambda\ p\bullet\ Fst_u\ (Fst_u\ p) \mapsto_u Snd_u\ (Snd_u\ p))$
$(Sep_u$
$(g \times_u f)$
$(\lambda\ p\bullet\ \exists\ q\ r\ s\bullet\ p = (q \mapsto_u r) \mapsto_u r \mapsto_u s))$

$\boldsymbol{Fun_u}$ $\vdash \forall\ x$
$\bullet\ Fun_u\ x$
$\Leftrightarrow Rel_u\ x$
$\wedge (\forall\ s\ t\ u$
$\bullet\ s \mapsto_u u \in_u x \wedge s \mapsto_u t \in_u x \Rightarrow u = t)$

$\boldsymbol{\nrightarrow_u}$ $\vdash \forall\ s\ t\bullet\ s \nrightarrow_u t = Sep_u\ (s \leftrightarrow_u t)\ Fun_u$

$\boldsymbol{\rightarrow_u}$ $\vdash \forall\ s\ t\bullet\ s \rightarrow_u t = Sep_u\ (s \nrightarrow_u t)\ (\lambda\ r\bullet\ s \subseteq_u Dom_u\ r)$

$\boldsymbol{\rightarrow_u\_closed}$ $\vdash \forall\ s$
$\bullet \rightarrow_u\_closed\ s$
$\Leftrightarrow (\forall\ d\ c\bullet\ d \in_u s \wedge c \in_u s \Rightarrow d \rightarrow_u c \in_u s)$

$\boldsymbol{\lambda_u}$ $\vdash \forall\ f\ s$
$\bullet\ \$\lambda_u\ f\ s$
$= Sep_u$
$(s \times_u Imagep_u\ f\ s)$
$(\lambda\ p\bullet\ Snd_u\ p = f\ (Fst_u\ p))$

$\boldsymbol{u}$ $\vdash \forall\ f\ x$
$\bullet\ f\ {}_u\ x$
$= (if\ \exists\ y\bullet\ x \mapsto_u y \in_u f$
$then\ \epsilon\ y\bullet\ x \mapsto_u y \in_u f$
$else\ f)$

$\boldsymbol{Id_u}$ $\vdash \forall\ s$
$\bullet\ Id_u\ s = Sep_u\ (s \times_u s)\ (\lambda\ x\bullet\ Fst_u\ x = Snd_u\ x)$

$\boldsymbol{\oplus_u}$ $\vdash \forall\ s\ t$
$\bullet\ s \oplus_u t$
$= Sep_u$
$(s \cup_u t)$
$(\lambda\ x$
$\bullet\ if\ Fst_u\ x \in_u Dom_u\ t$
$then\ x \in_u t$
$else\ x \in_u s)$

8.6 Theorems

$\boldsymbol{Set_u \mapsto_u\_thm}$ $\vdash \forall\ s\ t\bullet\ Set_u\ (s \mapsto_u t)$

$\boldsymbol{\mapsto_u\_eq\_thm}$ $\vdash \forall\ s\ t\ u\ v\bullet\ s \mapsto_u t = u \mapsto_u v \Leftrightarrow s = u \wedge t = v$

$\boldsymbol{Pair_u\_\in_u\_\mapsto_u\_thm}$
$\vdash \forall\ s\ t\bullet\ Pair_u\ s\ t \in_u s \mapsto_u t$

$\boldsymbol{Pair_u\_\in_u\_Gx_u\_\mapsto_u\_thm}$
$\vdash \forall\ s\ t\bullet\ Pair_u\ s\ t \in_u Gx_u\ (s \mapsto_u t)$

$\boldsymbol{\mapsto_u\_\in_u\_Gx_u\_Pair_u\_thm}$
$\vdash \forall\ s\ t\bullet\ s \mapsto_u t \in_u Gx_u\ (Pair_u\ s\ t)$

$\boldsymbol{\neg\mapsto_u\varnothing_u\_thm}$ $\vdash \forall\ x\ y\bullet\ \neg\ x \mapsto_u y = \varnothing_u$

$\boldsymbol{\neg\varnothing_u\mapsto_u\_thm}$ $\vdash \forall\ x\ y\bullet\ \neg\ \varnothing_u = x \mapsto_u y$

$\boldsymbol{GClose_u\mapsto_u\_thm}$
$\vdash \forall\ g$
$\bullet\ galaxy_u\ g$

$\Rightarrow (\forall\ s\ t \bullet\ s \in_u g \land t \in_u g \Rightarrow s \mapsto_u t \in_u g)$

tc*$\in_u$*_*$\mapsto_u$*_left_thm

$\vdash \forall\ s\ t \bullet\ s \in_u{}^+ s \mapsto_u t$

tc*$\in_u$*_*$\mapsto_u$*_right_thm

$\vdash \forall\ s\ t \bullet\ t \in_u{}^+ s \mapsto_u t$

$\mapsto_u$*_tc_thm* $\vdash \forall\ x\ y \bullet\ tc\ \$\in_u x\ (x \mapsto_u y) \land tc\ \$\in_u y\ (x \mapsto_u y)$

Rel*$_u$*_*$\varnothing_u$*_thm

$\vdash Rel_u\ \varnothing_u$

Set$_{uDomu}$_thm

$\vdash \forall\ r \bullet\ Set_u\ (Dom_u\ r)$

Dom$_u$_$\varnothing_u$_thm

$\vdash Dom_u\ \varnothing_u = \varnothing_u$

Dom$_u$_thm $\vdash \forall\ r\ y \bullet\ y \in_u Dom_u\ r \Leftrightarrow (\exists\ x \bullet\ y \mapsto_u x \in_u r)$

Dom$_u$_Gx$_u$_thm

$\vdash \forall\ r \bullet\ Dom_u\ r \in_u Gx_u\ r$

GClose$_u$_Dom$_u$_thm

$\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow (\forall\ r \bullet\ r \in_u g \Rightarrow Dom_u\ r \in_u g)$

tc$\in_u$_Dom$_u$_thm

$\vdash \forall\ x\ y \bullet\ x \in_u{}^+ Dom_u\ y \Rightarrow x \in_u{}^+ y$

Set$_{uRanu}$_thm

$\vdash \forall\ s \bullet\ Set_u\ (Ran_u\ s)$

Ran$_u$_$\varnothing_u$_thm

$\vdash Ran_u\ \varnothing_u = \varnothing_u$

Ran$_u$_thm $\vdash \forall\ r\ y \bullet\ y \in_u Ran_u\ r \Leftrightarrow (\exists\ x \bullet\ x \mapsto_u y \in_u r)$

GClose$_u$_Ran$_u$_thm

$\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow (\forall\ r \bullet\ r \in_u g \Rightarrow Ran_u\ r \in_u g)$

tc$\in_u$_Ran$_u$_thm

$\vdash \forall\ x\ y \bullet\ x \in_u{}^+ Ran_u\ y \Rightarrow x \in_u{}^+ y$

Dom$_u$_RanMap$_u$_thm

$\vdash \forall\ f\ r \bullet\ Rel_u\ r \Rightarrow Dom_u\ (RanMap_u\ f\ r) = Dom_u\ r$

Set$_{uFieldu}$_thm

$\vdash \forall\ r \bullet\ Set_u\ (Field_u\ r)$

Field$_u$_$\varnothing_u$_thm

$\vdash Field_u\ \varnothing_u = \varnothing_u$

tc$\in_u$_Field$_u$_thm

$\vdash \forall\ x\ y \bullet\ x \in_u{}^+ Field_u\ y \Rightarrow x \in_u{}^+ y$

Set$_{uRel2DepTypeu}$_thm

$\vdash \forall\ r \bullet\ Set_u\ (Rel2DepType_u\ r)$

Set$_{uDepSumu}$_thm

$\vdash \forall\ r \bullet\ Set_u\ (DepSum_u\ r)$

Set$_u$$\times_u$_thm $\vdash \forall\ s\ t \bullet\ Set_u\ (s \times_u t)$

$\times_u$*_spec* $\vdash \forall\ s\ t\ e$

$\bullet\ e \in_u s \times_u t$

$\Leftrightarrow (\exists\ l\ r \bullet\ l \in_u s \land r \in_u t \land e = l \mapsto_u r)$

f$\mapsto_{us}$_thm $\vdash \forall\ s\ t\ p \bullet\ p \in_u s \times_u t \Rightarrow Fst_u\ p \mapsto_u Snd_u\ p = p$

v$\in_u$$\times_u$_thm $\vdash \forall\ p\ s\ t$

$\bullet\ p \in_u s \times_u t \Rightarrow Fst_u\ p \in_u s \land Snd_u\ p \in_u t$

$\mapsto_u$$\in_u$$\times_u$*_thm*

$\vdash \forall\ l\ r\ s\ t \bullet\ l \mapsto_u r \in_u s \times_u t \Leftrightarrow l \in_u s \land r \in_u t$

$\leftrightarrow_u$$\subseteq_u$$\times_u$*_thm*

$\vdash \forall\ s\ t\ r \bullet\ r \in_u s \leftrightarrow_u t \Leftrightarrow Set_u\ r \land r \subseteq_u s \times_u t$

$\boldsymbol{\varnothing_u \in_u \leftrightarrow_u\_thm}$

$\vdash \forall\ s\ t \bullet\ \varnothing_u \in_u s \leftrightarrow_u t$

$\boldsymbol{f \mapsto_{us}\_thm1}$ $\vdash \forall\ p\ r\ s\ t$

$\bullet\ p \in_u r \land r \in_u s \leftrightarrow_u t \Rightarrow Fst_u\ p \mapsto_u Snd_u\ p = p$

$\boldsymbol{\in_u \leftrightarrow_u\_thm}$ $\vdash \forall\ p\ r\ s\ t$

$\bullet\ p \in_u r \land r \in_u s \leftrightarrow_u t$

$\Rightarrow Fst_u\ p \in_u s \land Snd_u\ p \in_u t$

$\boldsymbol{o_u\_thm}$ $\vdash \forall\ f\ g\ x$

$\bullet\ x \in_u f\ o_u\ g$

$\Leftrightarrow (\exists\ q\ r\ s$

$\bullet\ q \mapsto_u r \in_u g \land r \mapsto_u s \in_u f \land x = q \mapsto_u s)$

$\boldsymbol{o_u\_thm2}$ $\vdash \forall\ f\ g\ x\ y$

$\bullet\ x \mapsto_u y \in_u f\ o_u\ g$

$\Leftrightarrow (\exists\ z \bullet\ x \mapsto_u z \in_u g \land z \mapsto_u y \in_u f)$

$\boldsymbol{o_u\_Rel_u\_thm}$

$\vdash \forall\ r\ s \bullet\ Rel_u\ r \land Rel_u\ s \Rightarrow Rel_u\ (r\ o_u\ s)$

$\boldsymbol{o_u\_associative\_thm}$

$\vdash \forall\ f\ g\ h \bullet\ (f\ o_u\ g)\ o_u\ h = f\ o_u\ g\ o_u\ h$

$\boldsymbol{Rel_u\_\subseteq_u\_cp\_thm}$

$\vdash \forall\ x \bullet\ Rel_u\ x \Leftrightarrow (\exists\ s\ t \bullet\ x \subseteq_u s \times_u t)$

$\boldsymbol{Fun_u\_\varnothing_u\_thm}$

$\vdash Fun_u\ \varnothing_u$

$\boldsymbol{o_u\_Fun_u\_thm}$

$\vdash \forall\ f\ g \bullet\ Fun_u\ f \land Fun_u\ g \Rightarrow Fun_u\ (f\ o_u\ g)$

$\boldsymbol{Ran_u\_o_u\_thm}$

$\vdash \forall\ f\ g \bullet\ Ran_u\ (f\ o_u\ g) \subseteq_u Ran_u\ f$

$\boldsymbol{Dom_u\_o_u\_thm}$

$\vdash \forall\ f\ g \bullet\ Dom_u\ (f\ o_u\ g) \subseteq_u Dom_u\ g$

$\boldsymbol{Dom_u\_o_u\_thm2}$

$\vdash \forall\ f\ g$

$\bullet\ Ran_u\ g \subseteq_u Dom_u\ f \Rightarrow Dom_u\ (f\ o_u\ g) = Dom_u\ g$

$\boldsymbol{Fun_u\_RanMap_u\_thm}$

$\vdash \forall\ f\ g \bullet\ Fun_u\ g \Rightarrow Fun_u\ (RanMap_u\ f\ g)$

$\boldsymbol{\varnothing_u \in_u \nrightarrow_u\_thm}$

$\vdash \forall\ s\ t \bullet\ \varnothing_u \in_u s \nrightarrow_u t$

$\boldsymbol{\exists \nrightarrow_u\_thm}$ $\vdash \forall\ s\ t \bullet\ \exists\ f \bullet\ f \in_u s \nrightarrow_u t$

$\boldsymbol{\varnothing_u \in_u \varnothing_u \rightarrow_u\_thm}$

$\vdash \forall\ s\ t \bullet\ \varnothing_u \in_u \varnothing_u \rightarrow_u t$

$\boldsymbol{\exists \rightarrow_u\_thm}$ $\vdash \forall\ t \bullet\ (\exists\ x \bullet\ x \in_u t) \Rightarrow (\forall\ s \bullet\ \exists\ f \bullet\ f \in_u s \rightarrow_u t)$

$\boldsymbol{app\_thm1}$ $\vdash \forall\ f\ x \bullet\ (\exists_1\ y \bullet\ x \mapsto_u y \in_u f) \Rightarrow x \mapsto_u f\ {}_u\ x \in_u f$

$\boldsymbol{app\_thm2}$ $\vdash \forall\ f\ x\ y \bullet\ Fun_u\ f \land x \mapsto_u y \in_u f \Rightarrow f\ {}_u\ x = y$

$\boldsymbol{app\_thm3}$ $\vdash \forall\ f\ x \bullet\ Fun_u\ f \land x \in_u Dom_u\ f \Rightarrow x \mapsto_u f\ {}_u\ x \in_u f$

$\boldsymbol{o_u\_{}_u\_thm}$ $\vdash \forall\ f\ g\ x$

$\bullet\ Fun_u\ f$

$\land\ Fun_u\ g$

$\land\ x \in_u Dom_u\ g$

$\land\ Ran_u\ g \subseteq_u Dom_u\ f$

$\Rightarrow (f\ o_u\ g)\ {}_u\ x = f\ {}_u\ g\ {}_u\ x$

$\boldsymbol{app\_in\_Ran\_thm}$

$\vdash \forall\ x\ i \bullet\ Fun_u\ i \land x \in_u Dom_u\ i \Rightarrow i\ {}_u\ x \in_u Ran_u\ i$

$\boldsymbol{{}_u \in_u\_thm}$ $\vdash \forall\ f\ s\ t\ u$

$\bullet\ f \in_u s \nrightarrow_u t \wedge u \in_u Dom_u\ f \Rightarrow f\ _u\ u \in_u t$

$\boldsymbol{_u\in_u\_thm1}$ $\vdash \forall\ f\ s\ t\ u \bullet\ f \in_u s \rightarrow_u t \wedge u \in_u s \Rightarrow f\ _u\ u \in_u t$

$\boldsymbol{\in_u \nrightarrow_u \Rightarrow \in_u \rightarrow_u\_thm}$

$\vdash \forall\ f\ s\ t\ u \bullet\ f \in_u s \nrightarrow_u t \Rightarrow f \in_u Dom_u\ f \rightarrow_u t$

$\boldsymbol{Id_u\_thm1}$ $\vdash \forall\ s\ x \bullet\ x \in_u Id_u\ s \Leftrightarrow (\exists\ y \bullet\ y \in_u s \wedge x = y \mapsto_u y)$

$\boldsymbol{Id_u\_ap\_thm}$ $\vdash \forall\ s\ x \bullet\ x \in_u s \Rightarrow Id_u\ s\ _u\ x = x$

$\boldsymbol{Id_u \in_u \nrightarrow_u\_thm1}$

$\vdash \forall\ s\ t\ u \bullet\ s \subseteq_u t \cap_u u \Rightarrow Id_u\ s \in_u t \nrightarrow_u u$

$\boldsymbol{Id_u \in_u \nrightarrow_u\_thm2}$

$\vdash \forall\ s\ t\ u \bullet\ s \subseteq_u t \Rightarrow Id_u\ s \in_u t \nrightarrow_u t$

$\boldsymbol{Id_u\_clauses}$ $\vdash \forall\ s$

$\bullet\ Rel_u\ (Id_u\ s)$

$\wedge\ Fun_u\ (Id_u\ s)$

$\wedge\ Dom_u\ (Id_u\ s) =_u s$

$\wedge\ Ran_u\ (Id_u\ s) =_u s$

$\boldsymbol{\in_u \oplus_u\_thm}$ $\vdash \forall\ s\ t\ x$

$\bullet\ x \in_u s \oplus_u t$

$\Leftrightarrow (if\ Fst_u\ x \in_u Dom_u\ t$

$then\ x \in_u t$

$else\ x \in_u s)$

$\boldsymbol{\mapsto_u \in_u \oplus_u\_thm}$

$\vdash \forall\ s\ t\ x\ y$

$\bullet\ x \mapsto_u y \in_u s \oplus_u t$

$\Leftrightarrow x \mapsto_u y \in_u t \vee \neg\ x \in_u Dom_u\ t \wedge x \mapsto_u y \in_u s$

$\boldsymbol{\oplus_u\_Rel_u\_thm}$

$\vdash \forall\ s\ t \bullet\ Rel_u\ s \wedge Rel_u\ t \Rightarrow Rel_u\ (s \oplus_u t)$

$\boldsymbol{\oplus_u\_Fun_u\_thm}$

$\vdash \forall\ s\ t \bullet\ Fun_u\ s \wedge Fun_u\ t \Rightarrow Fun_u\ (s \oplus_u t)$

9 The Theory gsu-ord

9.1 Parents

$gsu\!-\!ax$

9.2 Children

GSU $gsu\!-\!seq$ $gsu\!-\!nat$

9.3 Constants

$\boldsymbol{SetOfSets_u}$	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{Connected_u}$	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{Ordinal_u}$	$'a\ GSU \rightarrow BOOL$
$\$<_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\$\leqslant_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\boldsymbol{Suc_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Successor_u}$	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{LimitOrdinal_u}$	
	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{Continuous_u}$	$('a\ GSU \rightarrow 'a\ GSU) \rightarrow BOOL$
$\$\boldsymbol{Ub_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\boldsymbol{Sup_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\$\boldsymbol{Sub_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow BOOL$
$\boldsymbol{Ssup_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Rank_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\$+_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{OrdMap_u}$	$('a\ GSU \rightarrow 'a\ GSU) \rightarrow BOOL$
$\$--_{\boldsymbol{u}}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$

9.4 Fixity

Right Infix 200:

$\boldsymbol{Sub_u}\boldsymbol{Ub_u}$

Right Infix 240:

$<_{\boldsymbol{u}}$ $\leqslant_{\boldsymbol{u}}$

Right Infix 300:

$+_{\boldsymbol{u}}$ $--_{\boldsymbol{u}}$

9.5 Definitions

$\boldsymbol{SetOfSets_u}$ $\vdash \forall\ s$
$\bullet\ SetOfSets_u\ s \Leftrightarrow Set_u\ s \land (\forall\ t\bullet\ t \in_u s \Rightarrow Set_u\ t)$

$\boldsymbol{Connected_u}$ $\vdash \forall\ s$
$\bullet\ Connected_u\ s$
$\Leftrightarrow (\forall\ t\ u$
$\bullet\ t \in_u s \land u \in_u s \Rightarrow t \in_u u \lor t = u \lor u \in_u t)$

$\boldsymbol{Ordinal_u}$ $\vdash \forall\ s$
$\bullet\ Ordinal_u\ s$
$\Leftrightarrow SetOfSets_u\ s \land Transitive_u\ s \land Connected_u\ s$

$<_u$ $\vdash \forall\ x\ y\bullet\ x <_u y \Leftrightarrow Ordinal_u\ x \land Ordinal_u\ y \land x \in_u y$

$\leqslant_u$ $\vdash \forall\ x\ y$
$\bullet\ x \leqslant_u y$
$\Leftrightarrow Ordinal_u\ x \land Ordinal_u\ y \land (x \in_u y \lor x = y)$

$\boldsymbol{Suc_u}$ $\vdash \forall\ x\bullet\ Suc_u\ x = x \cup_u Unit_u\ x$

$\boldsymbol{Successor_u}$ $\vdash \forall\ s\bullet\ Successor_u\ s \Leftrightarrow (\exists\ t\bullet\ Ordinal_u\ t \land s = Suc_u\ t)$

$\boldsymbol{LimitOrdinal_u}$
$\vdash \forall\ s$
$\bullet\ LimitOrdinal_u\ s$
$\Leftrightarrow Ordinal_u\ s \land \neg\ Successor_u\ s \land \neg\ s = \varnothing_u$

$\boldsymbol{Continuous_u}$ $\vdash \forall\ f$
$\bullet\ Continuous_u\ f$
$\Leftrightarrow (\forall\ x\bullet\ LimitOrdinal_u\ x \Rightarrow f\ x = ClIm_u\ f\ x)$

$\boldsymbol{Ub_u}$ $\vdash \forall\ \alpha\ \beta\bullet\ \alpha\ Ub_u\ \beta \Leftrightarrow (\forall\ \gamma\bullet\ \gamma \in_u \alpha \Rightarrow \gamma \leqslant_u \beta)$

$\boldsymbol{Sup_u}$ $\vdash \forall\ \alpha\bullet\ Sup_u\ \alpha = \bigcup_u \alpha$

$\boldsymbol{Sub_u}$ $\vdash \forall\ \alpha\ \beta\bullet\ \alpha\ Sub_u\ \beta \Leftrightarrow (\forall\ \gamma\bullet\ \gamma \in_u \alpha \Rightarrow \gamma <_u \beta)$

$\boldsymbol{Ssup_u}$ $\vdash \forall\ \alpha\bullet\ Ssup_u\ \alpha = \bigcup_u (Imagep_u\ Suc_u\ \alpha)$

$\boldsymbol{Rank_u}$ $\vdash \forall\ x\bullet\ Rank_u\ x = \bigcup_u (Imagep_u\ (Suc_u\ o\ Rank_u)\ x)$

$+_u$ $\vdash \forall\ \alpha\ \beta$
$\bullet\ \alpha +_u \beta$
$= (if\ \beta =_u \varnothing_u$
$then\ set_u\ \alpha$
$else\ ClIm_u\ (\$+_u\ \alpha)\ \beta)$

$\boldsymbol{OrdMap_u}$ $\vdash \forall\ f$
$\bullet\ OrdMap_u\ f \Leftrightarrow (\forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow Ordinal_u\ (f\ \alpha))$

$--_u$ $\vdash T$

9.6 Theorems

$\boldsymbol{Set_u\_ord_u\_thm}$
$\vdash \forall\ s\bullet\ Ordinal_u\ s \Rightarrow Set_u\ s$

$\boldsymbol{gsu\_ordinal\_ext\_thm}$
$\vdash \forall\ s\ t$
$\bullet\ Ordinal_u\ s \land Ordinal_u\ t$
$\Rightarrow (s = t \Leftrightarrow (\forall\ e\bullet\ e \in_u s \Leftrightarrow e \in_u t))$

$\boldsymbol{tc\in_u\_ord_u\_thm}$
$\vdash \forall\ \alpha\ \beta\bullet\ Ordinal_u\ \alpha \land \beta \in_u^+ \alpha \Rightarrow \beta \in_u \alpha$

$\boldsymbol{\in_u\_lt_u\_thm}$ $\vdash \forall\ \alpha\ \beta\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \land \alpha \in_u \beta \Rightarrow \alpha <_u \beta$

$\boldsymbol{lt_u\_\in_u\_thm}$ $\vdash \forall\ \alpha\ \beta\bullet\ \alpha <_u \beta \Rightarrow Ordinal_u\ \alpha \land Ordinal_u\ \beta \land \alpha \in_u \beta$

$\boldsymbol{\in_u\_\Leftrightarrow\_lt_u\_thm}$

$\vdash \forall\ \alpha\ \beta$
$\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta \Rightarrow (\alpha \in_u \beta \Leftrightarrow \alpha <_u \beta)$

tc$\in_u$_$\Leftrightarrow$_lt$_u$_thm
$\vdash \forall\ \alpha\ \beta$
$\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta \Rightarrow (\alpha \in_u{}^+\ \beta \Leftrightarrow \alpha <_u \beta)$

ord$_u$_$\in_u$_$\subset_u$_thm
$\vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow (\forall\ \beta\bullet\ \beta \in_u \alpha \Rightarrow \beta \subset_u \alpha)$

ord$_u$_$\in_u$_trans_thm
$\vdash \forall\ \alpha$
$\bullet\ Ordinal_u\ \alpha \Rightarrow (\forall\ \beta\ \gamma\bullet\ \beta \in_u \alpha\ \wedge\ \gamma \in_u \beta \Rightarrow \gamma \in_u \alpha)$

lt$_u$_$\subset_u$_thm $\vdash \forall\ \alpha\ \beta\bullet\ \alpha <_u \beta \Rightarrow \alpha \subset_u \beta$

lt$_u$_trans_thm
$\vdash \forall\ \alpha\ \beta\ \gamma\bullet\ \alpha <_u \beta\ \wedge\ \beta <_u \gamma \Rightarrow \alpha <_u \gamma$

$\leqslant_u$_lt$_u$_thm $\vdash \forall\ \alpha\ \beta$
$\bullet\ Ordinal_u\ \alpha\ \wedge\ Ordinal_u\ \beta$
$\Rightarrow (\alpha \leqslant_u \beta \Leftrightarrow \alpha <_u \beta\ \vee\ \alpha = \beta)$

$\leqslant_u$_lt$_u$_thm2
$\vdash \forall\ x\ y$
$\bullet\ x \leqslant_u y$
$\Leftrightarrow Ordinal_u\ x\ \wedge\ Ordinal_u\ y\ \wedge\ (x <_u y\ \vee\ x = y)$

$\leqslant_u$_$\subseteq_u$_thm $\vdash \forall\ \alpha\ \beta\bullet\ \alpha \leqslant_u \beta \Rightarrow \alpha \subseteq_u \beta$

$\leqslant_u$_trans_thm
$\vdash \forall\ \alpha\ \beta\ \gamma\bullet\ \alpha \leqslant_u \beta\ \wedge\ \beta \leqslant_u \gamma \Rightarrow \alpha \leqslant_u \gamma$

$\leqslant_u$_lt$_u$_trans_thm
$\vdash \forall\ \alpha\ \beta\ \gamma\bullet\ \alpha \leqslant_u \beta\ \wedge\ \beta <_u \gamma \Rightarrow \alpha <_u \gamma$

lt$_u$_$\leqslant_u$_trans_thm
$\vdash \forall\ \alpha\ \beta\ \gamma\bullet\ \alpha <_u \beta\ \wedge\ \beta \leqslant_u \gamma \Rightarrow \alpha <_u \gamma$

Set$_u$_Suc$_u$_thm
$\vdash \forall\ s\bullet\ Set_u\ (Suc_u\ s)$

$\subseteq_u$_Suc$_u$_thm
$\vdash \forall\ s\bullet\ s \subseteq_u Suc_u\ s\ \wedge\ Unit_u\ s \subseteq_u Suc_u\ s$

$\in_u$_Suc$_u$_thm
$\vdash \forall\ x\ y\bullet\ x \in_u Suc_u\ y \Leftrightarrow x \in_u y\ \vee\ x = y$

$\in_u$_Suc$_u$_thm2
$\vdash \forall\ s\bullet\ s \in_u Suc_u\ s$

ord$_u$_$\varnothing_u$_thm
$\vdash Ordinal_u\ \varnothing_u$

ord$_u$_eq_$\varnothing_u$_thm
$\vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha\ \wedge\ \alpha =_u \varnothing_u \Rightarrow \alpha = \varnothing_u$

tran$_u$_suc$_u$_tran$_u$_thm
$\vdash \forall\ x\bullet\ Transitive_u\ x \Rightarrow Transitive_u\ (Suc_u\ x)$

conn$_u$_suc$_u$_conn$_u$_thm
$\vdash \forall\ x\bullet\ Connected_u\ x \Rightarrow Connected_u\ (Suc_u\ x)$

ord$_u$_suc$_u$_ord$_u$_thm
$\vdash \forall\ x\bullet\ Ordinal_u\ x \Rightarrow Ordinal_u\ (Suc_u\ x)$

$\varnothing_u$_not_Suc$_u$_thm
$\vdash \neg\ (\exists\ \alpha\bullet\ Suc_u\ \alpha = \varnothing_u)$

$\neg$_eq_suc$_u$_thm
$\vdash \forall\ \alpha\bullet\ \neg\ \alpha = Suc_u\ \alpha$

$\leqslant_u$_suc$_u$_thm
$\vdash \forall\ \alpha\bullet\ Ordinal_u\ \alpha \Rightarrow \alpha \leqslant_u Suc_u\ \alpha$

$\boldsymbol{lt_u\_suc_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow \alpha <_u Suc_u\ \alpha$

$\boldsymbol{conn\_\subseteq_u\_conn}$

$\vdash \forall\ x \bullet\ Connected_u\ x \Rightarrow (\forall\ y \bullet\ y \subseteq_u x \Rightarrow Connected_u\ y)$

$\boldsymbol{conn\_\in_u\_ord}$ $\vdash \forall\ x \bullet\ Ordinal_u\ x \Rightarrow (\forall\ y \bullet\ y \in_u x \Rightarrow Connected_u\ y)$

$\boldsymbol{tran_u\_\in_u\_ord}$

$\vdash \forall\ x \bullet\ Ordinal_u\ x \Rightarrow (\forall\ y \bullet\ y \in_u x \Rightarrow Transitive_u\ y)$

$\boldsymbol{setofsets\_\in_u\_ord}$

$\vdash \forall\ x \bullet\ Ordinal_u\ x \Rightarrow (\forall\ y \bullet\ y \in_u x \Rightarrow SetOfSets_u\ y)$

$\boldsymbol{ord_u\_\in_u\_ord_u\_thm}$

$\vdash \forall\ x \bullet\ Ordinal_u\ x \Rightarrow (\forall\ y \bullet\ y \in_u x \Rightarrow Ordinal_u\ y)$

$\boldsymbol{ord_u\_tc\in_u\_ord_u\_thm}$

$\vdash \forall\ x \bullet\ Ordinal_u\ x \Rightarrow (\forall\ y \bullet\ y \in_u{}^+\ x \Rightarrow Ordinal_u\ y)$

$\boldsymbol{ord_u\_ext\_thm}$

$\vdash \forall\ \alpha\ \beta$
$\bullet\ Ordinal_u\ \alpha \wedge Ordinal_u\ \beta$
$\Rightarrow (\alpha = \beta \Leftrightarrow (\forall\ \gamma \bullet\ \gamma <_u \alpha \Leftrightarrow \gamma <_u \beta))$

$\boldsymbol{GClose_u\_Suc_u\_thm}$

$\vdash \forall\ g \bullet\ galaxy_u\ g \Rightarrow (\forall\ x \bullet\ x \in_u g \Rightarrow Suc_u\ x \in_u g)$

$\boldsymbol{tran_u\_\cap_u\_thm}$

$\vdash \forall\ x\ y$
$\bullet\ Transitive_u\ x \wedge Transitive_u\ y$
$\Rightarrow Transitive_u\ (x \cap_u y)$

$\boldsymbol{tran_u\_\cup_u\_thm}$

$\vdash \forall\ x\ y$
$\bullet\ Transitive_u\ x \wedge Transitive_u\ y$
$\Rightarrow Transitive_u\ (x \cup_u y)$

$\boldsymbol{\varnothing_u\_\leqslant_u\_thm}$ $\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow \varnothing_u \leqslant_u \alpha$

$\boldsymbol{\varnothing_u\_eq\_lt_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow \alpha = \varnothing_u \vee \varnothing_u <_u \alpha$

$\boldsymbol{\varnothing_u\_neq\_\in_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \wedge \neg\ \alpha = \varnothing_u \Rightarrow \varnothing_u \in_u \alpha$

$\boldsymbol{\varnothing_u\_neq_u\_\in_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \wedge \neg\ \alpha =_u \varnothing_u \Rightarrow \varnothing_u \in_u \alpha$

$\boldsymbol{conn\_\cap_u\_thm}$ $\vdash \forall\ x\ y$
$\bullet\ Connected_u\ x \wedge Connected_u\ y$
$\Rightarrow Connected_u\ (x \cap_u y)$

$\boldsymbol{setofsets\_\cap_u\_thm}$

$\vdash \forall\ x\ y$
$\bullet\ SetOfSets_u\ x \wedge SetOfSets_u\ y$
$\Rightarrow SetOfSets_u\ (x \cap_u y)$

$\boldsymbol{ord_u\_\cap_u\_thm}$

$\vdash \forall\ x\ y$
$\bullet\ Ordinal_u\ x \wedge Ordinal_u\ y \Rightarrow Ordinal_u\ (x \cap_u y)$

$\boldsymbol{trichot_u\_lemma}$

$\vdash \forall\ x\ y$
$\bullet\ Ordinal_u\ x \wedge Ordinal_u\ y \wedge x \subseteq_u y \wedge \neg\ x = y$
$\Rightarrow x \in_u y$

$\boldsymbol{trich\_for\_ord_u\_thm}$

$\vdash \forall\ x\ y$
$\bullet\ Ordinal_u\ x \wedge Ordinal_u\ y$

$\Rightarrow x <_u y \lor x = y \lor y <_u x$

$\boldsymbol{\subseteq_u\_\leqslant_u\_thm}$ $\vdash \forall x\ y \bullet Ordinal_u\ x \land Ordinal_u\ y \Rightarrow (x \subseteq_u y \Leftrightarrow x \leqslant_u y)$

$\boldsymbol{\subseteq_u\_\leqslant_u\_thm1}$ $\vdash \forall x\ y \bullet Ordinal_u\ x \land Ordinal_u\ y \land x \subseteq_u y \Rightarrow x \leqslant_u y$

$\boldsymbol{lt_u\_\Leftrightarrow\_\subset_u\_thm}$

$\vdash \forall \alpha\ \beta \bullet Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow (\alpha <_u \beta \Leftrightarrow \alpha \subset_u \beta)$

$\boldsymbol{\subset_u\_\Leftrightarrow\_lt_u\_thm}$

$\vdash \forall x\ y \bullet Ordinal_u\ x \land Ordinal_u\ y \Rightarrow (x \subset_u y \Leftrightarrow x <_u y)$

$\boldsymbol{ord_u\_sub_u\_\in_u\_thm}$

$\vdash \forall x\ y \bullet Ordinal_u\ x \land Ordinal_u\ y \land x \subset_u y \Rightarrow x \in_u y$

$\boldsymbol{\subset_u\_lt_u\_thm1}$

$\vdash \forall x\ y \bullet Ordinal_u\ x \land Ordinal_u\ y \land x \subset_u y \Rightarrow x <_u y$

$\boldsymbol{Successor_u\_ord_u\_thm}$

$\vdash \forall x \bullet Successor_u\ x \Rightarrow Ordinal_u\ x$

$\boldsymbol{wf\_lt_u\_thm}$ $\vdash well\_founded\ \$<_u$

$\boldsymbol{ord_u\_kind\_thm}$

$\vdash \forall n \bullet Ordinal_u\ n \Rightarrow n = \varnothing_u \lor Successor_u\ n \lor LimitOrdinal_u\ n$

$\boldsymbol{ord_u\_limit\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (\bigcup_u \alpha)$

$\boldsymbol{ord_u\_Sup_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (Sup_u\ \alpha)$

$\boldsymbol{Sup_u\_lUb_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow \alpha\ Ub_u\ Sup_u\ \alpha \land (\forall \gamma \bullet Ordinal_u\ \gamma \land \alpha\ Ub_u\ \gamma \Rightarrow Sup_u\ \alpha \leqslant_u \gamma)$

$\boldsymbol{Ssup_u\_ord_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (Ssup_u\ \alpha)$

$\boldsymbol{Ssup_u\_\in_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow (\forall \beta \bullet \beta \in_u \alpha \Rightarrow \beta \in_u Ssup_u\ \alpha)$

$\boldsymbol{Ssup_u\_lt_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow (\forall \beta \bullet \beta \in_u \alpha \Rightarrow \beta <_u Ssup_u\ \alpha)$

$\boldsymbol{Ssup_u\_\subseteq_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow \alpha\ Sub_u\ Ssup_u\ \alpha$

$\boldsymbol{Ssup_u\_TrCl_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ssup_u\ \alpha = TrCl_u\ \alpha$

$\boldsymbol{TrCl_u\_ord_u\_thm}$

$\vdash \forall \alpha \bullet (\forall \beta \bullet \beta \in_u \alpha \Rightarrow Ordinal_u\ \beta) \Rightarrow Ordinal_u\ (TrCl_u\ \alpha)$

$\boldsymbol{Set_u\_plus_u\_thm}$

$\vdash \forall\ \alpha\ \beta \bullet\ Set_u\ (\alpha +_u \beta)$

$\boldsymbol{ord_u\_set_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow set_u\ \alpha = \alpha$

$\boldsymbol{ord_u\_eq_u\_\varnothing_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \land \alpha =_u \varnothing_u \Rightarrow \alpha = \varnothing_u$

$\boldsymbol{ord_u\_plus_u\_thm}$

$\vdash \forall\ \alpha\ \beta$

$\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow Ordinal_u\ (\alpha +_u \beta)$

$\boldsymbol{plus_u\_\varnothing_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Set_u\ \alpha \Rightarrow \alpha +_u \varnothing_u = \alpha$

$\boldsymbol{plus_u\_\varnothing_u\_thm2}$

$\vdash \forall\ \alpha \bullet\ \alpha +_u \varnothing_u =_u \alpha$

$\boldsymbol{plus_u\_ur\_thm}$

$\vdash \forall\ \alpha\ x \bullet\ Set_u\ \alpha \land x =_u \varnothing_u \Rightarrow \alpha +_u x = \alpha$

$\boldsymbol{plus_u\_ur\_thm2}$

$\vdash \forall\ \alpha\ x \bullet\ x =_u \varnothing_u \Rightarrow \alpha +_u x =_u \alpha$

$\boldsymbol{plus_u\_ur\_thm3}$

$\vdash \forall\ \alpha\ x \bullet\ x =_u \varnothing_u \Rightarrow \alpha +_u x = set_u\ \alpha$

$\boldsymbol{plus_u\_\varnothing_u\_thm3}$

$\vdash \forall\ \alpha \bullet\ \alpha +_u \varnothing_u = set_u\ \alpha$

$\boldsymbol{plus_u\_\varnothing_u\_thm4}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow \alpha +_u \varnothing_u = \alpha$

$\boldsymbol{ne_u\_\varnothing_u\_set_u\_thm}$

$\vdash \forall\ x \bullet\ \neg\ x =_u \varnothing_u \Rightarrow set_u\ x = x$

$\boldsymbol{\varnothing_u\_plus_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow \varnothing_u +_u \alpha = \alpha$

$\boldsymbol{ClCo_u\_plus_u\_thm}$

$\vdash \forall\ \alpha \bullet\ ClCo_u\ (\$+_u\ \alpha)$

$\boldsymbol{plus_u\_eq\_\varnothing_u\_thm}$

$\vdash \forall\ \alpha\ \beta \bullet\ \alpha +_u \beta =_u \varnothing_u \Rightarrow \alpha =_u \varnothing_u \land \beta =_u \varnothing_u$

$\boldsymbol{\in_u\_ClIm_u\_plus_u\_thm}$

$\vdash \forall\ x\ \alpha\ t$

$\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ t$

$\Rightarrow (x \in_u ClIm_u\ (\$+_u\ \alpha)\ t$

$\Leftrightarrow (\exists\ y$

$\bullet\ y \in_u t \land (x = \alpha +_u y \lor x \in_u \alpha +_u y)))$

$\boldsymbol{plus_u\_def\_thm}$

$\vdash \forall\ \alpha\ \beta$

$\bullet\ Ordinal_u\ \beta \Rightarrow \alpha +_u \beta = \alpha \cup_u ClIm_u\ (\$+_u\ \alpha)\ \beta$

$\boldsymbol{ClIm_u\_ord_u\_thm}$

$\vdash \forall\ f\ \alpha$

$\bullet\ OrdMap_u\ f \land Ordinal_u\ \alpha \Rightarrow Ordinal_u\ (ClIm_u\ f\ \alpha)$

$\boldsymbol{lt_u\_ClIm_u\_thm}$

$\vdash \forall\ \alpha\ f\ \beta$

$\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \land OrdMap_u\ f$

$\Rightarrow (\alpha <_u ClIm_u\ f\ \beta \Leftrightarrow (\exists\ \delta \bullet\ \delta <_u \beta \land \alpha \leqslant_u f\ \delta))$

$\boldsymbol{OrdMap_u\_plus_u\_thm}$

$\vdash \forall\ \alpha \bullet\ Ordinal_u\ \alpha \Rightarrow OrdMap_u\ (\$+_u\ \alpha)$

$\boldsymbol{plus_u\_def\_thm2}$

$\vdash \forall\ \beta\ \alpha\ \gamma$

$\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta$

$\Rightarrow (\gamma <_u \alpha +_u \beta$

$\Leftrightarrow \gamma <_u \alpha \lor (\exists\ \delta \bullet\ \delta <_u \beta \land \gamma = \alpha +_u \delta))$

plus*$_u$*_Suc*$_u$*_thm

$\vdash \forall\ \alpha\ \beta$

$\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta$

$\Rightarrow \alpha +_u Suc_u\ \beta = Suc_u\ (\alpha +_u \beta)$

plus*$_u$*_mono_right_thm

$\vdash \forall\ \alpha\ \beta \bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \Rightarrow \alpha \leqslant_u \alpha +_u \beta$

plus*$_u$*_mono_right_thm2

$\vdash \forall\ \alpha\ \beta\ \gamma \bullet\ Ordinal_u\ \gamma \land \alpha <_u \beta \Rightarrow \gamma +_u \alpha <_u \gamma +_u \beta$

plus*$_u$*_assoc_thm

$\vdash \forall\ \gamma\ \alpha\ \beta$

$\bullet\ Ordinal_u\ \alpha \land Ordinal_u\ \beta \land Ordinal_u\ \gamma$

$\Rightarrow \alpha +_u \beta +_u \gamma = (\alpha +_u \beta) +_u \gamma$

10 The Theory gsu-nat

10.1 Parents

$gsu{-}ord$

10.2 Children

GSU

10.3 Constants

$\boldsymbol{natural\_number_u}$	$'a\ GSU \to BOOL$
$\$<_{\boldsymbol{un}}$	$'a\ GSU \to 'a\ GSU \to BOOL$
$\boldsymbol{Nat_u}$	$\mathbb{N} \to 'a\ GSU$

10.4 Fixity

Right Infix 240:

$<_{\boldsymbol{un}}$

10.5 Definitions

$\boldsymbol{natural\_number_u}$

$$\vdash \forall\ s \bullet natural\_number_u\ s \Leftrightarrow s = \emptyset_u \lor Successor_u\ s \land (\forall\ t \bullet\ t \in_u s \Rightarrow t = \emptyset_u \lor Successor_u\ t)$$

$<_{\boldsymbol{un}}$

$$\vdash \forall\ x\ y \bullet x <_{un} y \Leftrightarrow natural\_number_u\ x \land natural\_number_u\ y \land x \in_u y$$

$\boldsymbol{Nat_u}$

$$\vdash Nat_u\ 0 = \emptyset_u \land (\forall\ n \bullet Nat_u\ (n + 1) = Suc_u\ (Nat_u\ n))$$

10.6 Theorems

wf_nat$_u$_thm $\vdash$ *well_founded* $\$<_{un}$

nat$_u$_induct_thm

$\vdash \forall\ s\bullet\ (\forall\ x\bullet\ (\forall\ y\bullet\ y <_{un} x \Rightarrow s\ y) \Rightarrow s\ x) \Rightarrow (\forall\ x\bullet\ s\ x)$

nat$_u$_induct_thm2

$\vdash \forall\ p$
$\quad \bullet\ (\forall\ x$
$\qquad \bullet\ natural\_number_u\ x \wedge (\forall\ y\bullet\ y <_{un} x \Rightarrow p\ y)$
$\qquad\quad \Rightarrow p\ x)$
$\qquad \Rightarrow (\forall\ x\bullet\ natural\_number_u\ x \Rightarrow p\ x)$

ord$_u$_nat$_u$_thm

$\vdash \forall\ n\bullet\ natural\_number_u\ n \Rightarrow Ordinal_u\ n$

$\in_u$***_nat$_u$_ord$_u$_thm***

$\vdash \forall\ n$
$\quad \bullet\ natural\_number_u\ n \Rightarrow (\forall\ m\bullet\ m \in_u n \Rightarrow Ordinal_u\ m)$

nat$_u$_not_lim_thm

$\vdash \forall\ n\bullet\ natural\_number_u\ n \Rightarrow \neg\ LimitOrdinal_u\ n$

nat$_u$_zero_or_suc$_u$_thm

$\vdash \forall\ n\bullet\ natural\_number_u\ n \Rightarrow Successor_u\ n \vee n = \emptyset_u$

$\in_u$***_nat$_u$_not_lim_thm***

$\vdash \forall\ m\ n$
$\quad \bullet\ natural\_number_u\ n \wedge m \in_u n \Rightarrow \neg\ LimitOrdinal_u\ m$

$\in_u$***_nat$_u$_nat$_u$_thm***

$\vdash \forall\ n$
$\quad \bullet\ natural\_number_u\ n$
$\qquad \Rightarrow (\forall\ m\bullet\ m \in_u n \Rightarrow natural\_number_u\ m)$

nat$_u$_in_G$\emptyset_u$_thm

$\vdash \forall\ n\bullet\ natural\_number_u\ n \Rightarrow n \in_u Gx_u\ \emptyset_u$

ω_u_exists_thm

$\vdash \exists\ w\bullet\ \forall\ z\bullet\ z \in_u w \Leftrightarrow natural\_number_u\ z$

ord$_u$_nat$_u$_thm2

$\vdash \forall\ n\bullet\ Ordinal_u\ (Nat_u\ n)$

not_suc$_u$_nat$_u$_zero_thm

$\vdash \forall\ n\bullet\ \neg\ Suc_u\ (Nat_u\ n) = \emptyset_u$

lt$_u$_sum_thm $\vdash \forall\ x\ y\bullet\ x \leqslant y \Rightarrow (\exists\ z\bullet\ x + z = y)$

nat$_u$_mono_thm

$\vdash \forall\ x\ y\bullet\ Nat_u\ x \leqslant_u Nat_u\ (x + y)$

nat$_u$_one_one_thm

$\vdash \forall\ x\ y\bullet\ Nat_u\ x = Nat_u\ y \Rightarrow x = y$

nat$_u$_one_one_thm2

$\vdash \forall\ x\ y\bullet\ Nat_u\ x = Nat_u\ y \Leftrightarrow x = y$

11 The Theory gsu-seq

11.1 Parents

$gsu\!-\!fun \quad gsu\!-\!ord$

11.2 Children

GSU

11.3 Constants

$\boldsymbol{Seq_u}$	$'a\ GSU \rightarrow BOOL$
$\boldsymbol{Length_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Head_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Front_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Back_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{Tail_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\$\boldsymbol{@_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{UnitSeq_u}$	$'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{SeqCons_u}$	$'a\ GSU \rightarrow 'a\ GSU \rightarrow 'a\ GSU$
$\boldsymbol{SeqDisp_u}$	$'a\ GSU\ LIST \rightarrow 'a\ GSU$

11.4 Fixity

Right Infix 300:

$\boldsymbol{@_u}$

11.5 Definitions

$\boldsymbol{Seq_u}$	$\vdash \forall\ s\bullet\ Seq_u\ s \Leftrightarrow Fun_u\ s \wedge Ordinal_u\ (Dom_u\ s)$
$\boldsymbol{Length_u}$	$\vdash Length_u = Dom_u$
$\boldsymbol{Head_u}$	$\vdash \forall\ s\bullet\ Head_u\ s = s\ {}_u\ \varnothing_u$
$\boldsymbol{Front_u}$	$\vdash \forall\ \alpha\ s\bullet\ Front_u\ \alpha\ s = \alpha \lhd_u s$
$\boldsymbol{Back_u}$	$\vdash \forall\ \alpha\ s\bullet\ Back_u\ \alpha\ s = s\ o_u\ (\lambda_u\ \beta\bullet\ \beta +_u \alpha)\ (Dom_u\ s)$
$\boldsymbol{Tail_u}$	$\vdash \forall\ s\bullet\ Tail_u\ s = Back_u\ (Suc_u\ \varnothing_u)\ s$
$\boldsymbol{@_u}$	$\vdash \forall\ s\ t\bullet\ s\ @_u\ t = s \cup_u t$
$\boldsymbol{UnitSeq_u}$	$\vdash \forall\ e\bullet\ UnitSeq_u\ e = \varnothing_u \mapsto_u e$
$\boldsymbol{SeqCons_u}$	$\vdash \forall\ e\ s\bullet\ SeqCons_u\ e\ s = UnitSeq_u\ e\ @_u\ s$
$\boldsymbol{SeqDisp_u}$	$\vdash SeqDisp_u\ [] = \varnothing_u$ $\wedge\ (\forall\ e\ s$ $\bullet\ SeqDisp_u\ (Cons\ e\ s) = SeqCons_u\ e\ (SeqDisp_u\ s))$

11.6 Theorems

$\boldsymbol{Seq_u\_RanMap_u\_thm}$

$\vdash \forall\ f\ s\bullet\ Seq_u\ s \Rightarrow Seq_u\ (RanMap_u\ f\ s)$

12 The Theory GSU

12.1 Parents

$gsu{-}seq$ $gsu{-}nat$ $gsu{-}ord$ $gsu{-}fun$ $gsu{-}ax$

12.2 Children

misc3

13 INDEX