Proof Contexts

Makarius Wenzel
TU München

August 2009

Aspects of locality

Locality means . . .

- working relatively to a *context*
  (proof environment or local theory)
- replacing logical encodings by *native elements* of Isabelle/Isar
- moving results between contexts via *morphisms*
  e.g. from abstract theory to concrete application

Consequences:

- improved flexibility and scalability
- simplified construction and composition of add-on tools
- block-structured operations, instead of fiddling with variables

Proof context elements

```
{
  fix $x$
  have $B\ x$  $\langle proof \rangle$
}
note $\langle \bigwedge x.\ B\ x \rangle$
```

```
{
  def $x \equiv a$
  have $B\ x$  $\langle proof \rangle$
}
note $\langle B\ a \rangle$
```

```
{
  assume $A$
  have $B$  $\langle proof \rangle$
}
note $\langle A \Longrightarrow B \rangle$
```

```
{
  obtain $a$ where $B\ a$  $\langle proof \rangle$
  have $C$  $\langle proof \rangle$
}
note $\langle C \rangle$
```

Examples

See `Slides2/Ex1.thy`

Clausal statements

Universal clauses: fixes x **assumes** $A\ x$ **shows** $B\ x$
based on primitive Isar context elements

Existential clauses: obtains a **where** $B\ a$ **|** $\ldots$ expands to
fixes thesis **assumes** $\bigwedge a.\ B\ a \Longrightarrow \mathit{thesis}$ **and** $\ldots$ **shows** thesis

Examples:

theorem
  assumes $\exists\, x.\ B\ x$
  obtains a **where** $B\ a$

theorem
  assumes $A \lor B$
  obtains $(\mathit{left})\ A \mid (\mathit{right})\ B$

theorem
  assumes $A \land B$
  obtains A **and** B

theorem
  fixes $x\ y :: \mathit{nat}$
  obtains $(\mathit{lt})\ x < y \mid (\mathit{eq})\ x = y \mid (\mathit{gt})\ x > y$

Generic context data

Internally record of data-slots (dynamically typed disjoint sums)

Programming interface recovers strongly static typing

functor $ProofDataFun(ARGS)$: $RESULT$, where

$ARGS$ = *sig type* T *val* $init\colon theory \to T$ *end*

$RESULT$ = *sig val* $get\colon context \to T$ *val* $map\colon (T \to T) \to context \to context$ *end*

Example content:

- Logical declarations (variables, assumptions)
- Definitions (terms, theorems)
- Type-inference information
- Syntax annotations (mixfix grammar)
- Hints for proof tools (simpset, claset, arithmetic setup etc.)

Examples

See `Slides2/Ex2.thy`