

Specification of Red JonPRL Signatures

Jon Sterling

January 10, 2016

1 Grammar

The grammar of Red JonPRL signatures is presented in Figure 1. Note that an optional production of sort s is formatted $\langle s \rangle$ in the rules.

$sigexp$	$::=$	$\langle \cdot \rangle$	empty signature
		$sigexp. sigdec$	signature extension
$sigdec$	$::=$	$\text{Def } opid \langle [params] \rangle \langle (args) \rangle : sortid = [term]$	operator definition
		$\text{Tac } opid \langle [params] \rangle \langle (args) \rangle = [tac]$	tactic definition
		$\text{Thm } opid \langle [params] \rangle \langle (args) \rangle : [term] \text{ by } [tac]$	theorem declaration
$params$	$::=$	$\langle \cdot \rangle$	empty parameter list
		$params, symbind$	parameter list extension
$args$	$::=$	$\langle \cdot \rangle$	empty argument list
		$args, metabind$	argument list extension
$symbind$	$::=$	$symid : sortid$	symbol binding
$metabind$	$::=$	$metaid : valence$	metavariable binding
$valence$	$::=$	$\langle \{ [sortlist] \} \langle [sortlist] \rangle . \rangle sortid$	valence
$sortlist$	$::=$	$\langle \cdot \rangle$	empty sort list
		$sortlist, sortid$	sort list extension

Figure 1: Grammar of signature expressions. The identifier sorts $opid$, $sortid$, $symid$ and $metaid$ can be assumed to be arbitrary strings; the sorts $term$ and tac are left uninterpreted.