



Jekejeke Minlog Installation

Version 1.0.3, December 24th, 2018



XLOG Technologies GmbH

Jekejeke Prolog

Minimal Logic 1.0.3

Installation Guide

Author: XLOG Technologies GmbH
Jan Burse
Freischützgasse 14
8004 Zürich
Switzerland

Date: December 24th, 2018
Version: 0.30

Participants: None

Warranty & Liability

To the extent permitted by applicable law and unless explicitly otherwise agreed upon, XLOG Technologies GmbH makes no warranties regarding the provided information. XLOG Technologies GmbH assumes no liability that any problems might be solved with the information provided by XLOG Technologies GmbH.

Rights & License

All industrial property rights regarding the information - copyright and patent rights in particular - are the sole property of XLOG Technologies GmbH. If the company was not the originator of some excerpts, XLOG Technologies GmbH has at least obtained the right to reproduce, change and translate the information.

Reproduction is restricted to the whole unaltered document. Reproduction of the information is only allowed for non-commercial uses. Small excerpts can be used if properly cited. Citations must at least include the document title, the product family, the product version, the company, the date and the page. Example:

... Defined predicates with arity>0, both static and dynamic, are indexed on the functor of their first argument [1, p.17] ...

[1] Language Reference, Jekejeke Prolog 0.8.1, XLOG Technologies GmbH, Switzerland, February 22nd, 2010

Trademarks

Jekejeke is a registered trademark of XLOG Technologies GmbH.

Table of Contents

1	Introduction	7
2	Releases Notes 1.0.....	8
2.1	Release 1.0.3	8
2.2	Release 1.0.1	9
2.3	Release 1.0.0	9
3	Releases Notes 0.9.....	10
3.1	Release 0.9.5	10
3.2	Release 0.9.4	10
3.3	Release 0.9.3	11
3.4	Release 0.9.2	11
3.5	Release 0.9.1	11
4	Releases Notes 0.8.....	12
4.1	Release 0.8.8	12
4.2	Release 0.8.7	12
4.3	Release 0.8.6	13
4.4	Release 0.8.5	13
4.5	Release 0.8.4	13
4.6	Release 0.8.3	14
4.7	Release 0.8.2	15
4.8	Release 0.8.1	15
5	Releases Notes 0.7.....	16
5.1	Release 0.7.9	16
5.2	Release 0.7.8	17
5.3	Release 0.7.6	17
5.4	Release 0.7.5	18
5.5	Release 0.7.4	18
5.6	Release 0.7.3	19
5.7	Release 0.7.1	19
6	Releases Notes 0.6.....	20
6.1	Release 0.6.6	20
6.2	Release 0.6.5	21
6.3	Release 0.6.4	21
6.4	Release 0.6.3	22
6.5	Release 0.6.2	22
6.6	Release 0.6	23
6.7	Release 0.5	23
7	Extension Installation	24
7.1	Hotspot Installation	25
7.2	Dalvik Installation.....	27
8	Support Files.....	29
8.1	Documentation	29
8.2	Example Sources	30
8.3	Interpreter Sources.....	30
9	Known Issues	31
9.1	Extension Issues	31
9.2	Installation Issues	31
	Pictures	32
	Tables	32
	References.....	32

Change History

Jan Burse, June 15th, 2011, 0.1:

- Initial version.

Jan Burse, July 19th, 2012, 0.2:

- First public version.

Jan Burse, August 8th, 2012, 0.3:

- Dalvik installation instructions introduced.

Jan Burse, December 10th, 2012, 0.4:

- Release notes for 0.6.2 added.

Jan Burse, February 20th, 2013, 0.5:

- Release notes for 0.6.3 added and system requirements sections move.

Jan Burse, May 23th, 2013, 0.6:

- Release notes for 0.6.4 added.

Jan Burse, June 5th, 2013, 0.7:

- Release notes for 0.6.5 added.

Jan Burse, August 4th, 2013, 0.8:

- Release notes for 0.6.6 added and known issues sections moved.

Jan Burse, April 18th, 2014, 0.9:

- Release notes for 0.7.1 added.

Jan Burse, June 9th, 2014, 0.10:

- Release notes for 0.7.3 added.

Jan Burse, October 23rd, 2014, 0.11:

- Release notes for 0.7.4 added.

Jan Burse, February 11th, 2015, 0.12:

- Release notes for 0.7.5 added and known issues sections extended.

Jan Burse, May 28th, 2015, 0.13:

- Release notes for 0.7.6 added.

Jan Burse, August 9th, 2015, 0.14:

- Release notes for 0.7.8 added.

Jan Burse, September 2nd, 2015, 0.15:

- Release notes for 0.7.9 added.

Jan Burse, January 4th, 2016, 0.16:

- Release notes for 0.8.1 added.

Jan Burse, February 25th, 2016, 0.17:

- Release notes for 0.8.2 added.

Jan Burse, March 7th, 2016, 0.18:

- Release notes for 0.8.3 added.

Jan Burse, May 16th, 2016, 0.19:

- Release notes for 0.8.4 added.

Jan Burse, June 24th, 2016, 0.20:

- Release notes for 0.8.5 added.

Jan Burse, August 31th, 2016, 0.21:

- Release notes for 0.8.6 added.

Jan Burse, November 15th, 2016, 0.22:

- Release notes for 0.8.7 added.

Jan Burse, December 14th, 2016, 0.23:

- Release notes for 0.8.8 added.

Jan Burse, March 11th, 2017, 0.24:

- Release notes for 0.9.1 added.

Jan Burse, April 30th, 2017, 0.25:

- Release notes for 0.9.2 added.

Jan Burse, July 8th, 2017, 0.26:

- Release notes for 0.9.3 added.

Jan Burse, October 17th, 2017, 0.27:

- Release notes for 0.9.4 added.

Jan Burse, July 05th, 2017, 0.28:

- Release notes for 1.0.0 added.

Jan Burse, October 29th, 2018, 0.29:

- Release notes for 1.0.1 added.

Jan Burse, December 24th, 2018, 0.30:

- Release notes for 1.0.3 added.

1 Introduction

The Jekejeke Minlog module is available as a platform independent archive. Customers can also download documentation and samples. In the following we describe the download contents and its most basic use.

- **Release Notes 1.0:** This section lists the changes for the release family 1.0.
- **Release Notes 0.9:** This section lists the changes for the release family 0.9.
- **Release Notes 0.8:** This section lists the changes for the release family 0.8.
- **Release Notes 0.7:** This section lists the changes for the release family 0.7.
- **Release Notes 0.6:** This section lists the changes for the release family 0.6.
- **Extension Installation:** Here we describe how the Jekejeke Minlog capability can be installed and used.
- **Support Files:** Here we describe how the Jekejeke Minlog capability support files can be installed.

2 Releases Notes 1.0

This section lists the changes concerning the Jekejeke Minlog module, documentation and samples for the release family 1.0:

- [Release 1.0.3](#)
- [Release 1.0.1](#)
- [Release 1.0.0](#)

2.1 Release 1.0.3

The following features have been provided for the Jekejeke Minlog module of version 1.0.3:

Finite Domain

- New predicate `random_indomain/1` for CLP(FD) introduced.
- New predicate `random_label/1` for CLP(FD) introduced.
- New predicate `random_labeling/1` for CLP(B) introduced.
- .

2.2 Release 1.0.1

The following features have been provided for the Jekejeke Minlog module of version 1.0.1:

Language Reference

- Total rework of the module "hypo", now using ordinary goals as actions.
- New predicate (;)/3 for temporary alternatives in module "hypo" introduced.
- Total rework of the module "delta", now using ordinary goals as actions.
- The directive cosmetic/1 has been removed.
- The predicate property cosmetic has been removed.
- The delta computations predicates are now marked as sys_notrace.
- The set operator {}/1 to wrap condition goals has been removed.
- Non-annotated goals are now automatically condition goals.
- New module "chr" for constraint handling rules introduced.
- The forward chaining can now handle disjunctions (;)/2 in the condition.
- New module "asp" for answer set programming introduced.
- The chart parser can now handle disjunction (;)/2 in the body.
- The chart parser operator is now (==:)/2 to avoid clash with module "chr".

Finite Domain

- The CLP(FD) realization now uses new module "hypo".
- The CLP(FD) realization now uses new module "delta".

2.3 Release 1.0.0

The following features have been provided for the Jekejeke Minlog module of version 1.0.0:

Language Reference

- New predicate divmod/4 in module elem introduced.
- Predicate sys_test_bit/2 in module bits renamed to testbit/2.
- The evaluable function isqrt/1 of module elem now uses Zimmerman algorithm.
- New predicate sqrtrem/4 in module elem introduced.
- New evaluable function iroot/2 in module elem introduced.
- New predicate rootrem/4 in module elem introduced.

3 Releases Notes 0.9

This section lists the changes concerning the Jekejeke Minlog module, documentation and samples for the release family 0.9:

- [Release 0.9.5](#)
- [Release 0.9.4](#)
- [Release 0.9.3](#)
- [Release 0.9.2](#)
- [Release 0.9.1](#)

3.1 Release 0.9.5

The following features have been provided for the Jekejeke Minlog module of version 0.9.5:

Language Reference

- New predicate card/2 in SAT solver introduced.

3.2 Release 0.9.4

The following features have been provided for the Jekejeke Minlog module of version 0.9.4:

Language Reference

- New module state in package term introduced.
- New module clpb in package finite introduced.
- Fixed that del_atts/2 from module verify wasn't visible.
- Fixed that del_attr/2 from module unify wasn't visible.

3.3 Release 0.9.3

The following features have been provided for the Jekejeke Minlog module of version 0.9.3:

Language Reference

- Test cases for `sto/1` from module `herbrand` introduced.
- New module `unify` in package `term` introduced.
- New module `verify` in package `term` introduced.

Frequent Predicates

- New function `hipow/2` introduced in module `ring`.

3.4 Release 0.9.2

The following features have been provided for the Jekejeke Minlog module of version 0.9.2:

Language Reference

- Enhanced the predicates `bitcount/2` and `bitlength/2` to negative values.
- New test cases for bit properties predicates introduced.
- New predicate `isqrt/2` in module `elem` introduced.
- New test cases for number theoretic predicates introduced.

Frequent Predicates

- New module `ordered` in package `gauss` introduced.
- New module `ring` in package `gauss` introduced.
- New module `radical` in package `leibniz` introduced.
- New predicate `integ/3` in module `deriv` introduced.

3.5 Release 0.9.1

The following features have been provided for the Jekejeke Minlog module of version 0.9.1:

Language Reference

- New predicate `sys_freezer/1` in module `trail` of package `experiment` introduced.
- New predicate `lcm/3` in module `elem` of package `misc` introduced.

Frequent Predicates

- New package `groebner` for symbolic polynomials introduced.
- New package `gauss` for symbolic matrixes introduced.
- New package `leibniz` for automatic differentiation introduced.

4 Releases Notes 0.8

This section lists the changes concerning the Jekejeke Minlog module, documentation and samples for the release family 0.8:

- [Release 0.8.8](#)
- [Release 0.8.7](#)
- [Release 0.8.6](#)
- [Release 0.8.5](#)
- [Release 0.8.4](#)
- [Release 0.8.3](#)
- [Release 0.8.2](#)
- [Release 0.8.1](#)

4.1 Release 0.8.8

The following features have been provided for the Jekejeke Minlog module of version 0.8.8:

Language Reference

- Array subscripts can now access non ground compounds in CLP(FD).
- The low-level built-ins of package misc have been made open source.

Frequent Predicates

- First package decimal for multi precision functions introduced.

4.2 Release 0.8.7

The following features have been provided for the Jekejeke Minlog module of version 0.8.7:

Language Reference

- Missing documentation on `{}/1` in module hypo completed.
- Missing documentation on `{}/1` in module delta completed.

Finite Domain

- CLP(FD) test cases made to run with new test runner.
- CLP(FD) test cases now mirrored on GitHub as well.

4.3 Release 0.8.6

The following features have been provided for the Jekejeke Minlog module of version 0.8.6:

Language Reference

- Access to forward chaining facts from static predicates now forbidden.
- Modification of forward chaining facts from static predicates now forbidden.

4.4 Release 0.8.5

The following features have been provided for the Jekejeke Minlog module of version 0.8.5:

Finite Domain

- Replace call/2 by is/2 so that partial functions throw an exception.
- Replace call/2 by is/2 so that non-deterministic functions don't leave a choice point.

Term Domain

- Improved inference for neq/2 predicate now fully simulating unification.
- Improved inference for neq/2 predicate compact representation by module maps.

4.5 Release 0.8.4

The following features have been provided for the Jekejeke Minlog module of version 0.8.4:

Language Reference

- The character extension was removed, now available through module text in runtime.
- The strings extension was removed, now available in runtime.
- New evaluable function gcd/2 introduced in module elem.
- The chart parser now automatically places the forward annotation (+)/1.

Finite Domain

- The reified negation (#)/1 has now its own equation.
- Interaction of reified domain range with other domain ranges realized.
- Interaction of reified equality/inequality with other equality/inequality realized.
- In the label/1 predicate infinite domains are now filtered out and cantor paired.
- Interaction of multiplication constraint improved to always allow contraction.

4.6 Release 0.8.3

The following features have been provided for the Jekejeke Minlog module of version 0.8.3:

Language Reference

- The predicates `sys_recorda_hook/2` and `sys_recordz_hook/2` have been removed.
- The predicates `sys_erase_hook/2` and `sys_compiled_hook/2` have been removed.
- The predicate `sys_compile_hook/2` has been removed.
- New predicate `sys_compile_hook/3` introduced.
- The predicates `sys_assume_hook/1` and `sys_assume_hook/1` have been removed.
- The predicates `sys_assume_ref/2` and `sys_retire_ref/2` have been removed.
- New construct `(-)/1` in the hypothetical reasoner introduced.
- The predicates `embedded_abnormal/1` and `continuation_abnormal/1` removed.
- New predicates `hypo_abnormal/1` and `minus_abnormal/1` introduced.
- New predicates `sys_find_goal/3` and `find_abnormal/1` introduced.
- The predicates `sys_attr/1` and `sys_new_attr/1` have been removed.
- The predicate `sys_compile_hook/3` now promotes an ordinary variable.
- The predicate `sys_clause_hook/3` now fails on an ordinary variable.
- The predicate `sys_ensure_attr/1` has been removed.
- The predicate `sys_bind_serno/1` has been removed.
- New predicate `sys_ensure_serno/1` introduced.
- New predicate `sys_ripple/1` introduced.
- New miscellaneous definitions section in documentation.
- New indexes section in documentation.

Finite Domain

- Test cases for the core CLP(FD) constraint solver introduced.
- Different hooks for different constraint forms in the CLP(FD) constraint solver.
- Better serial number generation in the CLP(FD) constraint solver.
- New set operator intersection `(\)/2` and complement `(\)/1` introduced.
- New module `reify` introduced, automatically re-exported by the module `clpfd`.
- First set of test cases for the reified CLP(FD) constraint solver introduced.

4.7 Release 0.8.2

The following features have been provided for the Jekejeke Minlog module of version 0.8.2:

Language Reference

- New module `cont` to access the continuation queue introduced.
- New module `suspend` with predicates `freeze/2` and `when/2` introduced.
- New Prolog flag `sys_verify` introduced.
- The predicate `sys_undo/1` now remembers the `sys_verify` flag as well.
- The variable hooks are now executed with the `sys_verify` flag set to off.
- New predicate `sys_assume_cont/1` in module `assume` introduced.
- Longstanding problem in exception handling of attribute variables fixed.
- The module `diff` has been removed.
- New module `herbrand` with predicates `sto/1` and `neq/2` introduced.
- The module `clpfd` now uses the new constraint projection API.
- The module `herbrand` now uses the new constraint projection API.
- The module `suspend` now uses the new constraint projection API.

4.8 Release 0.8.1

The following features have been provided for the Jekejeke Minlog module of version 0.8.1:

Language Reference

- The CLP(FD) now also accepts closures in integer expressions.
- The CLP(FD) now also accepts closures in variable lists.

5 Releases Notes 0.7

This section lists the changes concerning the Jekejeke Minlog module, documentation and samples for the release family 0.7:

- [Release 0.7.9](#)
- [Release 0.7.8](#)
- [Release 0.7.6](#)
- [Release 0.7.5](#)
- [Release 0.7.4](#)
- [Release 0.7.3](#)
- [Release 0.7.1](#)

5.1 Release 0.7.9

The following features have been provided for the Jekejeke Minlog module of version 0.7.9:

Language Reference

- Predicate `sys_new_surrogate/1` removed from module `misc/struc`.

5.2 Release 0.7.8

The following features have been provided for the Jekejeke Minlog module of version 0.7.8:

Language Reference

- Module `system/ref` removed, now found in runtime library as `experiment/ref`.
- Predicate `sys_clause_expose/3` removed from module `minimal/delta`.
- Predicate `sys_unify_notrace/2` removed from module `minimal/delta`.
- The predicate `(=>)/2` doesn't use the flag `sys_barrier` anymore in `minimal/hypo`.
- New multi-file predicate `embedded_abnormal/1` introduced in `minimal/hypo`.
- The predicate `(<=)/1` doesn't use the flag `sys_barrier` anymore in `minimal/hypo`.
- New multi-file predicate `continuation_abnormal/1` introduced in `minimal/hypo`.
- Predicates `indomain/2` and `indomains/2` removed from module `finite/clpfd`.
- New predicate `indomain/1` introduced in module `finite/clpfd`.
- New function `abs/1` introduced in module `finite/clpfd`.

5.3 Release 0.7.6

The following features have been provided for the Jekejeke Minlog module of version 0.7.6:

Language Reference

- Error messages removed from documentation.
- Error messages are now multi-lingual, English and German.
- Error messages are now found on open source web site.
- The documentation has now a title page.

5.4 Release 0.7.5

The following features have been provided for the Jekejeke Minlog module of version 0.7.5:

Language Reference

- Display of CLP(FD) constraints is now idempotent.
- Fix a problem with exception handling in attribute variable hooks.

5.5 Release 0.7.4

The following features have been provided for the Jekejeke Minlog module of version 0.7.4:

Language Reference

- The chart parser terminal representation 'D'/3 removed.
- Terminal representation now works via 'D'/2 facts.
- The chart parser predicate postwords/2 removed.
- Posting words now works via hypothetical reasoning.
- Syntax operator (&&-->)/2 replaced by syntax operator (--:)/2.
- Syntax operator (&-->)/2 replaced by syntax operator (==>)/2.
- New Code adapted so that it uses package local visibility.
- Code adapted so that it uses structured module names.
- New package misc introduced.
- New package system introduced.
- The predicate stored/0 removed.
- The multi-file predicate sys_pretty_event/1 removed.
- The predicate sys_get_variable_names/1 removed.
- The current CLP(FD) constraints are now always shown in the top-level.
- The current diff/2 constraints are now always shown in the top-level.
- The predicate property_delta/0 removed.
- New predicate property_cosmetic/0 introduced.

5.6 Release 0.7.3

The following features have been provided for the Jekejeke Minlog module of version 0.7.3:

Language Reference

- The predicates `posted/0` and `posted/1` removed.
- Syntax operator `(&:-)/2` replaced by syntax operator `(<=)/2`.
- Forward chaining now uses new annotation syntax for bodies.
- Forward chaining now performs call-site transfer.
- Forward chaining now uses hypothetical reasoning syntax for assumptions.
- Forward chaining now uses hypothetical reasoning predicates to trigger.
- New predicate `sys_term_kernel/2` introduced.
- New predicate `sys_term_globals/2` introduced.
- Hypothetical reasoning now supports intra-clausal variables.
- The forward closure is not preloaded anymore.
- The forward closure turned into module `library(delta)`.
- The finite domain solver is now a true module.
- The chart parsing is not preloaded anymore.
- New module `library(assume)` introduced.
- New module `library(hypo)` introduced.
- New module `library(diff)` introduced.

5.7 Release 0.7.1

The following features have been provided for the Jekejeke Minlog module of version 0.7.1:

Language Reference

- Finite domain solver is not preloaded anymore.

6 Releases Notes 0.6

This section lists the changes concerning the Jekejeke Minlog module, documentation and samples for the release family 0.6:

- [Release 0.6.6](#)
- [Release 0.6.5](#)
- [Release 0.6.4](#)
- [Release 0.6.3](#)
- [Release 0.6.2](#)
- [Release 0.6](#)
- [Release 0.5](#)

6.1 Release 0.6.6

The following features have been provided for the Jekejeke Minlog module of version 0.6.6:

Language Reference

- The predicates `post/1` and `post/2` can now be advised.
- New operator `(&-)/1` introduced to mark goals for deletion.
- New operator `(&&)/2` introduced for asymmetric conjunction.
- New forward clause operator `(&:-)/2` introduced.
- New chart clause operators `(&-->)/2` and `(&&-->)/2` introduced.
- New predicates `sys_freeze_var/2` and `sys_melt_var/2` introduced.
- New predicate `sys_bound_var/1` introduced.
- Now obsolete predicates `compile_ref/3` and `sys_global_ref/2` removed.
- New predicates `sys_new_attr/1` and `sys_attr/1` introduced.
- New predicate `sys_bind_serno/1` introduced.
- New predicates `sys_compile_hook/2` and `sys_compiled_hook/2` introduced.
- New predicates `sys_recorda_hook/2` and `sys_recordz_hook/2` introduced.
- New predicates `sys_clause_hook/3` and `sys_erase_hook/2` introduced.
- New predicates `assume_hook/2` and `retire_hook/2` introduced.
- The finite domain constraints are now based on native Prolog variables.
- New interval calculation for the multiplication constraints introduced.
- New special cases for the multiplication constraints introduced.
- New interval calculation for the special cases introduced.
- The predicate `sys_access_ref/3` can now be advised.
- The predicates `sys_retire_refs/1` and `sys_assume_refs/1` can now be advised.
- The finite domain constraints now use native Prolog unification where permitted.

6.2 Release 0.6.5

The following features have been provided for the Jekejeke Minlog module of version 0.6.5:

Language Reference

- New operator (&-->)/2 introduced for chart DCGs.
- New predicate (;)/2 supported in forward chaining rules.
- New predicate (;)/2 supported in chart DCGs.
- New predicate last_atom_concat/3 introduced.
- New predicate last_sub_atom/5 and last_sub_atom/4 introduced.
- The predicates compile_ref/[2,3] now respect public/private.

6.3 Release 0.6.4

The following features have been provided for the Jekejeke Minlog module of version 0.6.4:

Language Reference

- The predicate sys_make_ref/1 replaced by the predicate sys_new_surrogate/1.
- The predicate compiled_ref/2 replaced by the predicate sys_compiled_ref/2.
- New predicate compile_ref/3 has been introduced.
- New predicate sys_global_ref/2 has been introduced.

6.4 Release 0.6.3

The following features have been provided for the Jekejeke Minlog module of version 0.6.3:

Language Reference

- The predicate `compile_ref/2` doesn't understand `(&)/2` and `unit/0` anymore.
- The predicate `compiled_ref/2` doesn't understand `(&)/2` and `unit/0` anymore.
- The predicate `sys_recorda_ref/1` doesn't understand `(&)/2` and `unit/0` anymore.
- The predicate `sys_recordz_ref/1` doesn't understand `(&)/2` and `unit/0` anymore.
- The predicate `sys_erase_ref/1` doesn't understand `(&)/2` and `unit/0` anymore.
- The predicate `assume_ref/1` doesn't understand `(&)/2` and `unit/0` anymore.
- The predicate `retire_ref/1` doesn't understand `(&)/2` and `unit/0` anymore.

6.5 Release 0.6.2

The following features have been provided for the Jekejeke Minlog module of version 0.6.2:

Language Reference

- Port cosmetics for the delta computation function introduced.
- New elementary evaluable function `ulp/1` introduced.
- New string evaluable functions `lowercase/1` and `uppercase/1` introduced.
- New clause reference predicates `assume_ref/2` and `retire_ref/2` introduced.
- New clause reference predicates `assume_ref/1` and `retire_ref/1` introduced.
- New forward closure predicate `post/2` introduced.
- New forward closure predicate `posted/1` introduced.
- Call-site transfer for building the bottom of a chart introduced.
- Call-site transfer for querying a chart introduced.

6.6 Release 0.6

The following features have been provided for the Jekejeke Minlog module of version 0.6:

Language Reference

- There is now a Hotspot and a Dalvik specific version available.

6.7 Release 0.5

We started the beta-testing campaign with this release. This is our first public release.

7 Extension Installation

The extension comes in two flavours. There is a version for the Hotspot Java virtual machine and a version for the Dalvik Java virtual machine.

In the following we will give more details on the installation of the different versions:

- [Hotspot Installation](#)
- [Dalvik Installation](#)

7.1 Hotspot Installation

A manual package is available for any Hotspot Java virtual machine. The package only includes the Jekejeke Minlog capability. You will download the following archive file:

```
module.jar # Top-Level and Embedding
```

You might copy the archive file to the destination directory `<dest>` of your choice. The archive file can be used for the following purposes:

- **Top-Level:** The archive file can be used together with a Prolog query answer loop.
- **Embedding:** The archive file can extend a Prolog system for application use.
- **System Requirements:** The system requirements of the actual version.

Top-Level

The archive file can be used together with a Prolog query answer loop. The archive can be either combined with the runtime library or the development environment. In the following we will show the combination with the runtime library. The combination with the development environment works similarly.

We start explaining the command line approach. As a first step the archive file has to be added to the class path. This means we cannot anymore invoke the runtime library with the `-jar` option. Instead we have to explicitly provide the main class name. The main class name will be the name of the toolkit. To initialize the minimal logic extension we need also to pass an `-e` option to the toolkit:

```
java -cp interpreter.jar;minlog.jar \  
    jekpro.platform.headless.ToolkitLibrary \  
    -e jekmin.platform.headless.CapabilityMinlog # on windows  
java -cp interpreter.jar:minlog.jar \  
    jekpro.platform.headless.ToolkitLibrary \  
    -e jekmin.platform.headless.CapabilityMinlog # on linux and mac
```

For the `-e` option we provide the Jekejeke Minlog capability. This capability would not be found if the archive file would not be in the class path. The `-e` option causes the toolkit to initialize the capability and thus to provide the functionality of the Jekejeke Minlog capability. The `-e` option also causes to prompt the end-user for the license.

One can alternatively use the settings dialog to both extend the class path and to set an `-e` option. These settings will then be stored in the user profile and recalled upon starting the runtime library. This will allow starting the runtime library via the `-jar` option and nevertheless have the Jekejeke Minlog capability available.

Last but not least all these steps can also be performed programmatically in a Prolog text already for the runtime library. The class path can be extended by the archive file via the system predicate `sys_add_path/1` and extension capability can be initialized via the system predicate `sys_init_capability/1`.

Embedding

The archive file can extend a Prolog system for application use. In most cases the top-level methods from the previous sections also work for application embedding. But it could be that the extension capability needs to be accessed at compile time. It is then necessary to add the archive file to the class path during compilation:

```
javac -cp interpreter.jar;module.jar;. <main>.java # on windows
javac -cp interpreter.jar:module.jar:. <main>.java # on linux and mac
```

The only Java programming object that is visible for the Jekejeke Minlog module is the class `CapabilityMinlog`. Currently this class does not export some constants or functions. It might be only useful for use in administration functions of the runtime library. When you want to execute your application that has been compiled with the Jekejeke Minlog module, you might use the following command line.

```
java -cp interpreter.jar;module.jar;. <main> # on windows
java -cp interpreter.jar:module.jar:. <main> # on linux and mac
```

For a recommendation on additional command line options see the runtime library installation document. Alternatively you can use an integrated development environment to compile and execute your Java class. All you probably have to do is create an appropriate project and then register all the archive files in the class path of the project.

You might also unpack the Jekejeke Minlog capability and include it in a `.jar` together with your compiled byte code and then execute this `.jar`.

Further you might want to deploy the Jekejeke Minlog capability together with your applets or servlets. In the case of applets this will not work, since we currently do not provide a mechanism to access profile data remotely and therefore the license will not be found. In case of servlets all you need to do is copy the archive file into the `WEB-INF/lib` directory. If the license has been activated on the server, for example by using the top level and the `-e` option from a shell, the Jekejeke Minlog capability should work.

System Requirements

The Jekejeke Minlog module of version 0.6.3 requires at least:

- Hotspot 1.5 [\[1\]](#)

The Jekejeke Minlog module of version 0.6.3 is compatible with at least:

- Jekejeke Prolog Runtime Library 0.9.8 [\[3\]](#)
- Jekejeke Prolog Development Environment 0.9.8 [\[3\]](#)

7.2 Dalvik Installation

Manual packages are available for any Dalvik/Android Java virtual machine. The packages only include the Jekejeke Prolog runtime library. You will download the following archive files:

```
module.apk # Top-Level
module.zip # Embedding
```

You might copy the archive file to the destination directory `<dest>` of your choice. The archive file can be used for the following purposes:

- **Top-Level:** The archive file can be used together with a Prolog query answer loop.
- **Embedding:** The archive file can extend a Prolog system for application use.
- **System Requirements:** The system requirements of the actual version.

Top-Level

The archive file can be used together with a Prolog query answer loop. The archive can be either combined with the runtime library or the development environment. In the following we will show the combination with the runtime library. The combination with the development environment works similarly.

We currently do not provide a command line for the Dalvik/Android platform. Therefore there is also no way to make the Jekejeke Minlog capability available via the command line on this platform. But the settings dialog of the runtime library will allow to extend the class path and to set an `-e` option. These settings will then be stored in the user profile and recalled upon starting the runtime library.

The settings will read as follows:

- **Class Path:** `/data/app/jekmin.platform.headless-1.apk`
- **Capability:** `jekmin.platform.headless.CapabilityMinlog`

The class path might differ from device to device. Also the number after the dash varies with the number of installation of the package on your device.

Last but not least all these steps can also be performed programmatically in a Prolog text already for the runtime library. The class path can be extended by the archive file via the system predicate `sys_add_path/1` and extension capability can be initialized via the system predicate `sys_init_capability/1`.

Embedding

The archive file can extend a Prolog system for application use. In most cases the top-level methods from the previous sections also work for application embedding. But it could be that the extension capability needs to be accessed at compile time. It is then necessary to add the archive file to the class path during compilation:

```
javac -bootclasspath android.jar \  
      -cp interpreter.zip;module.zip;. \  
      <activity>.java          # on windows  
javac -bootclasspath android.jar \  
      -cp interpreter.zip:module.zip;. \  
      <activity>.java          # on linux and mac
```

The only Java programming object that is visible for the Jekejeke Minlog module is the class `CapabilityMinlog`. Currently this class does not export some constants or functions. It might be only useful for use in administration functions of the runtime library.

After compilation you then need to execute the further steps that are necessary in the process of building an Android package are. Alternatively you can use an integrated development environment to compile and execute your Java class.

System Requirements

The Jekejeke Minlog module of version 0.6.3 requires at least:

- Dalvik 1.6 (API 4) [\[2\]](#)

The Jekejeke Minlog module of version 0.6.3 is compatible with at least:

- Jekejeke Prolog Runtime Library 0.9.8 [\[3\]](#)
- Jekejeke Prolog Development Environment 0.9.8 [\[3\]](#)

8 Support Files

Download of the support files is available for all platforms that have a ZIP extractor. The download includes the support files for the Jekejeke Minlog module. You will download the following archive file:

```
supmin.zip          # The support files archive
```

You can use a GUI tool or a command line tool of your choice that is able to deal with .zip files. If all else fails you can use the jar utility that comes with a Java development kit installation. The archive file can be extracted with the following jar utility command. Make sure that you are inside destination directory <dest>:

```
jar xf supmin.zip
```

After unpacking the archive one can easily explore its contents with a HTML browser.

The support files archive contains the following kind of support files:

- **Documentation:** The documentation for the Jekejeke Minlog module is provided as HTML split files or as full PDF documents.
- **Example Sources:** The source files for the Minlog module example programs are provided as ZIP archive files.
- **Interpreter Sources:** The partial source files for the Minlog module interpreter are provided as a ZIP archive files.

8.1 Documentation

The documentation for the Jekejeke Minlog module is provided as HTML split files or as full PDF documents. The HTML split files can be view with a HTML browser. To view the PDF files a PDF reader needs to be available.

The support files archive contains the following documentation:

```
15_min
+---- 10_docu
      +---- 02_reference      # Language Reference
      +---- 03_interface     # Programming Interface
      +---- 04_installation  # Installation Guide
```

The full PDF documents are located in the files called package.pdf in the above directories.

8.2 Example Sources

The source files for the Jekejeke Minlog module example programs are provided as source archive files. The source files mainly include Prolog texts and Java classes. But they might also include other types of artefacts.

The support files archive contains the following sources:

```
15_min
+---- 10_docu
      +---- 02_reference      # Language Reference
```

The source archive files are located in the files package.zip in the above directories.

You can easily run the programs by means of the Java command line or from within an integrated development environment.

8.3 Interpreter Sources

The partial source files for the Jekejeke Minlog module interpreter programs are provided as source archive files. The source files mainly include Prolog texts and Java classes. But they might also include other types of artefacts.

The support files archive contains the following sources:

```
15_min
+---- 02_reference      # Language Reference
```

The source archive files are located in the files package.zip in the above directories.

The sources are mainly there to give a more detailed documentation of the inner working of the interpreter. But the sources can also be used to create derivative work, except for special predicates, which currently come without source.

9 Known Issues

The end-user might check requested features or bug fixes against the list of known issues. We are constantly working on improving the Minlog extension. The issues are grouped as follows:

- [Extension Issues](#)
- [Installation Issues](#)

9.1 Extension Issues

The following issues are known for the Jekejeke Minlog module of version 0.8.2:

Language

- Missing universal quantifier based on variable type side condition.
- Missing universal quantifier based on variable occurrence side condition.
- Missing equality construct based on Clark equation theory, i.e. diff/2.
- Missing universal quantifier based for Clark equation theory, i.e. diff/2.
- Missing epistemic disjunction and Herbrand results for Clark completion.
- Missing disjunctive forward chaining variant for Clark completion.
- Hypothetical reasoning issue of reclaiming unbinds still unsolved.
- .

9.2 Installation Issues

The following issues are known for the Jekejeke Minlog module of version 0.8.3:

Installation

- .

Pictures

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tables

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

References

- [1] Java™ 2 Platform Standard Edition 5.0, Tiger, Sun Microsystems, 2004
<http://www.oracle.com/technetwork/java/javase/index-jsp-135232.html>
- [2] Android 1.6 Platform, Donut, Google Inc., September 2009
<http://developer.android.com/sdk/android-1.6.html>
- [3] Jekejeke Prolog Runtime Library, XLOG Technologies GmbH, Switzerland
http://www.jekejeke.ch/ide/ideprod/en/docs/05_run/package.html
- [4] Jekejeke Prolog Development Environment, XLOG Technologies GmbH, Switzerland
http://www.jekejeke.ch/ide/ideprod/en/docs/10_dev/package.html