Changes.doc

COLLABORATORS			
	TITLE : Changes.doc		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		December 7, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Changes.doc

1.1 "

```
   $RCSfile: Changes.doc $
Description: Changes and bug-fixes made to Oberon-A

 Created by: fjc (Frank Copeland)
  $Revision: 1.9 $
    $Author: fjc $
      $Date: 1995/07/02 16:56:50 $
```

```
~Release~1.6~~~~~~~~~~

~Release~1.5~~~~~~~~~~
~Release~1.5~update~1~
~Release~1.5~update~2~
~Release~1.5~update~3~

~Release~1.4~~~~~~~~~~
~Release~1.4~update~1~
~Release~1.4~update~2~

~Release~1.3~~~~~~~~~~
~Release~1.3~update~1~
~Release~1.3~update~2~

~Release~1.2~~~~~~~~~~

~Release~1.1~~~~~~~~~~
```

1.2 Release 1.1

```
Bugs fixed
----------

OC
  * Enforcer hit caused when no DST parameter was specified
```

```
    [A. Weinert].
  * Same error code used for different errors. [R. Waspe]

OL
  * The module name passed as a parameter was case-sensitive [R. Waspe].

OberonSys.lib
  * The garbage collector was mistreating SysBlks in the mark phase.
    Fireworks mode :-) [fjc]

Improvements made
-----------------

FPE
  * Added "Program-Create Directory" menu item and removed references
    to "Make Code" button in FPE.doc.

OC
  * Changed command line arguments:
    - Options now must come first;
    - Multiple filename arguments allowed.
  * Batch compiles implemented.
  * OLIB: is now the default symbol file search path.
  * Error files are output in the current directory with the name
    "<module>.err".
  * Compiles can be interrupted with CTRL-C.

OL
  * Command line arguments slightly changed.
  * Changed format of .with file to suit Commodore's ALink.
  * OLIB: is now the default symbol file search path.

ORU
  * Changed to generate a batch file to be used with the BATCH option of
    the compiler.
  * OLIB: is now the default symbol file search path.

Oberon-A/S
  * Oberon-Startup file containing useful aliases for Shell users
    created.

StdIO.mod
  * Added code to check for CTRL-C.
```

1.3 Release 1.2

```
Bugs fixed
----------

OC
  * Batch file was not closed when batch compile interrupted by CTRL-C.
    [fjc]
  * Numerous bugs in the translation of type-bound procedures,
    especially when forward declared.  It was a wonder they worked at
    all. [K. Juha, J. Hawkins]
```

```
Improvements made
-----------------

OC
  * "/" appended to paths not ending in "/" or ":".
  * Now checks for at least one RETURN statement in a function
    procedure. Generates code to check at run-time for a missing RETURN
    statement in a function procedure. Added "$r" switch to turn checks
    off.

OL
  * "/" appended to paths not ending in "/" or ":".

ORU
  * "/" appended to paths not ending in "/" or ":".

Lists.mod
  * Added FindNameNoCase().

Workbench.mod
  * Implemented AddAppIconA() using inline code to get around the
    problem with using A4 to pass parameters.
```

1.4 Release 1.3

```
Bugs fixed
----------

OC
  * Compiling non-Oberon files caused gurus due to writing error reports
    to a NIL filehandle. [Torsten ?]
  * Processing escaped characters in strings could lead to infinite
    loops. [A. Weinert]
  * Number of bytes of initialised data was understated. [fjc]
  * Failed to free registers when dereferencing BPointers. [fjc]
  * _Any_ constant could be assigned to a set variable. [fjc]

FPE
  * In at least two cases memory belonging to IntuiSup library was being
    accessed (read) _after_ the message was replied. [fjc]

Graphics.mod
  * Wrong register used for parameter of OpenFont(). [fjc]

OL
  * Program failed to free locks on symbol files if it was interrupted
    with CTRL-C.

ORU
  * Program failed to free locks on files if it was interrupted with
    CTRL-C.

Improvements made
-----------------
```

```
OC
  * Error #5 (end of file in comment) now reports the position of the
    start of the offending comment.
  * Removed all references to UNION types.  They were more more trouble
    than they were worth.
  * String literals seperated by whitespace are concatenated.
  * Implemented varargs for Amiga library calls.
  * Reorganised symbol table as a binary search tree, for a serious
    performance improvement.
  * New symbol file format produces smaller files and increases number
    of exportable types.

Module SYSTEM
  * Added TYPETAG type and procedures for processing type tags.
  * Added SETREG and REG procedures.

Amiga Interface
  * General overhaul:
    - Reorganised and renamed some modules:
      - Rexx.mod --> Rexx.mod and RexxSysLib.mod
      - KeyMaps.mod --> KeyMap.mod and KeyMapLib.mod
      - InputEvents.mod --> InputEvent.mod
      - Console.mod --> Console.mod and ConUnit.mod
      - FileSystem.mod --> FileSysRes.mod
      - Merged Types.mod with Exec.mod
    - Added missing modules:
      - MathLibrary.mod
      - MathIeeeSingBas.mod
      - MathIeeeSingTrans.mod
    - Made sure that the translation of C names into Oberon was done
      consistently:
      - Kept prefixes for constants except where they were just
        abbreviations of the module name.
      - Deleted prefixes for record fields except where this caused name
        clashes.
    - Fixed parameter types of LIBCALLs.
    - Created LIBCALL versions using varargs as well as TagItem arrays.
    - Created variant LIBCALLs with alternative parameter lists where
      appropriate.
    - Wrote new utility procedures:
      - ExecUtil.mod
      - BoopsiUtil.mod
      - HookUtil.mod

FPE
  * Changed source code to use new Amiga interfaces.
  * Minor details changed.

Examples
  * Wrote some (well, ported some from the RKM).

Files.mod
  * Module now keeps track of all files opened and closes any left
    open when the program exits.

OL
```

```
  * Added specific support for ALink and dlink.
```

1.5 Release 1.3 update 1

```
Bugs fixed
----------

OC
  * Calling a type-bound procedure through a de-referenced CPointer
    variable caused an address trap. [fjc]

Improvements made
-----------------

OC
  * Compiler now passes NIL when the actual parameter for an ARRAY OF
    CHAR LIBCALL parameter is an empty string.

Bugs introduced :-(
-------------------

OC
  * Strings longer than 1 character are not being passed to LIBCALL
    parameters.
```

1.6 Release 1.3 update 2

```
Bugs fixed
----------

OC
  * The string passing bug introduced in update 1 was fixed.
```

1.7 Release 1.4

```
Bugs fixed
----------

OC
  * The register involved in a LIBCALL parameter was being reserved too
    soon, causing register allocation errors in some cases where the
    actual parameters were expressions involving function procedures,
    or long integer or real arithmetic. [C.Ziegeler]
  * Fixing the above bug uncovered another, in which the parameter
    register was being freed before it was reserved. This only happened
    when the actual parameter was a record field referenced through a
    pointer, or an array element. [fjc]
  * It was possible to dereference a function procedure that returned a
    pointer type as if it were a pointer variable, with unpredictable
    results. [C.Ziegeler]
```

```
  * There was no check that forward declared procedures were actually
    implemented. The linker would have spotted this anyway. [fjc]
  * The stack offsets of procedure parameters were being written to the
    symbol file. The $L compiler switch changed these offsets, making
    the symbol file invalid. [fjc]
  * The column reported for error locations was incorrect if there were
    TAB characters in the line. [J.Ferreira]
  * Direct comparisons with TRUE gave incorrect results in some
    circumstances (specifically, when dealing with boolean return
    values from dos.library functions). [E.Dewald]

Intuition.mod
  * menuNull was mis-spelled nenuNull.

Exec.mod
  * paXXX flags declared incorrectly. [J.Ferreira]

Improvements made
-----------------

OEL
  * Oberon-A Error Lister, contributed by Johan Ferreira.

Module SYSTEM
  * The parameters to SYSTEM.SETCLEANUP are now a single assignable
    procedure. There is no need for a variable to hold the old cleanup
    procedure, or a return code parameter.
  * Added SYSTEM.RC to return the current return code.
  * SYSTEM.NEW now has an optional parameter for passing memory
    requirements.
  * Extended the range of types that can be used with bit operations
    (SYSTEM.LSH, etc.)

All
  * Changed to use new SYSTEM.SETCLEANUP format.

OC
  * Changed the $Z switch from a module switch to a global switch.
  * No longer generates multiple error reports at the same location.
  * Implemented foreign procedures.
  * Implemented the $A compiler switch.
  * SYSTEM.LONGWORD variables can now be assigned any value whose type
    <= 32 bits. The same for SYSTEM.WORD when the type is <= 16 bits.
    Integers are sign-extended, all other values are zero-extended.
  * Implemented NIL checking when dereferencing pointers, calling
    procedures from variables and executing type guards with pointers.
  * Changed register parameter declarations to use square brackets
    instead of braces.
  * Implemented stack checking.
  * Added TEXTERR command line option.
  * Changed to output binary error file by default.
  * Implemented $s compiler switch.
  * Changed error numbers.

FPE
  * arp.library is now only used if dos.library < V37 or asl.library
    cannot be opened.
```

```
OL
  * Added options to allow a linker to be called directly.

OD
  * Definition file utility created.

Amiga Interface
  * Updated to Release 40.15 (3.1).
    - Added new declarations to most existing modules.
    - Created new modules:
      - AmigaGuide.mod
      - Bullet.mod
      - CardRes.mod
      - CDDevice.mod
      - Datatypes.mod
      - Gadgets.mod
      - Locale.mod
      - LowLevel.mod
      - NonVolatile.mod
      - Prefs.mod
      - RealTime.mod
  * Changed modules to use foreign code interface:
    - BoopsiUtil.mod
    - RexxUtil.mod
    - Added the related object files (Classface.obj and rexxvars.o) to
      OberonSys.lib. Note that this is a temporary solution until a
      more complete foreign code interface is implemented.
  * Retained Commodore's comments in the source code.

Third-party library interface modules:
  * New modules contributed by various people:
    - GuiEnv.mod
    - GuiEnvSupport.mod
    - ReqTools.mod
    - TextFieldGadget.mod
    - UMS.mod

Dos.mod
  * Created SetVBufPtr() as an alias for SetVBuf(), to be used to pass
    NIL buffers.

Intuition.mod
  * Changed 'args' parameters for EasyRequest functions to
    SYS.LONGWORD.

Editor support
  * Macros and ARexx scripts to allow the compiler, linker, etc. to be
    called from within an editor. The only supported editor at the
    present is AmokEd, but the scripts should be adaptable to other
    editors.
```

1.8 Release 1.4 update 1

```
Bugs fixed
----------

OC
  * Bug in processing of formal parameters caused infinite loops if two
    parameters had the same name. [C.Ziegler]

Improvements made
-----------------

OC
  * Checks for the existence of directories given in SYM and DST
    arguments.

Revision control
  * Uses Johan Fereirra's OBumpRev to manage revision numbers for the
    compiler and utilities.

Oberon-A Library modules
  * Implemented module Kernel to replace functions currently in module
    SYSTEM that will be removed in Release 1.5.
  * Added ObjectExists() and DirExists() to module DosUtil.
  * Changed module ExecUtil to use functions from module Kernel.
  * Created module WbConsole to independantly handle creating a stdio
    environment for Workbench programs.
  * Changed module StdIO to use module WbConsole.
```

1.9 Release 1.4 update 2

```
Bugs fixed
----------

OC
  * Calling type-bound procedures from arrays of objects caused
    register allocation errors. [C.Grigis]
```

1.10 Release 1.5

```
Introduction
------------

Release 1.5 is a re-working of the entire Oberon-A package. There are
two major changes and many minor ones.

The first major change is that Oberon-A has been brought in line with
the recommendations contained in the Oakwood Report. This document was
the result of a meeting of Oberon developers in late 1992. It contains
guidelines for developers covering a number of areas, including:

  - Compiler control, including compiler options, pragmas and
    conditional compilation.
```

- External code interfaces.
- Standard library modules.
- Language extensions.

Unfortunately, the status of the Oakwood Report is now extremely doubtful, as ETH appears to have withdrawn its earlier endorsement of the document. It has a number of weaknesses, especially in the recommendations for external code interfaces and standard library modules. However, in the absence of any other guidelines, Oberon-A will continue to follow the Oakwood Report where practical.

The second major change is in the Amiga interface modules. These have been changed to bring them as close as possible to the interfaces used by the AmigaOberon compiler. This decision was made after a long discussion in the Oberon-A mailing list. The aim is to make it as simple as possible to re-use Amiga-specific code written for the AmigaOberon compiler. I decided that the many problems in the AmigaOberon interfaces were outweighed by the advantages of code portability and re-use.

Work will soon begin on a new set of Amiga interface modules, which will be as close as possible to the original C header files provided by Commodore. This will especially apply to the names of objects, which will be identical to those used in the header files. Ideally these new interfaces will be produced automatically by a translation program.

Other important changes include a Workbench interface for most programs, preferences settings and preferences tools for OC and OL, and localisation of most programs.

Bugs fixed

OC

* Dereferencing a pointer in an array caused register allocation problems when NIL checking was enabled.
* The standard function ABS was not implemented for REALs.
* If type-bound procedures were not declared in a particular order, they could be allocated to the wrong slots in the type descriptor.
* The potential existed for the use of the wrong addressing mode when accessing array elements.

OEL

* Attempted to dereference a NIL pointer if it encountered an error code not present in the catalog.

Changes made

OC

* Run-time system is now implemented in module Kernel.
* Replaced the old compiler switches with Oakwood-style pragmas and options.
* Implemented source code control (conditional compilation).

```
  * Completely new external code interface, including Amiga library
    calls and pointer variants.
  * Completely new ReadArgs() command line template.
  * Added Workbench interface.
  * Added preferences settings, including a preferences tool (OCPrefs)
  * Added localisation support for strings.
  * Overhauled module SYSTEM, removing defunct procedures and adding
    SYSTEM.CC.
  * The code generated for run-time checks now includes a pointer to
    the module's name and the current position in the source text.
  * See OC.doc and OCPrefs.doc for details.

OL

  * Added support for the new run-time system and external code
    interface.
  * Completely new ReadArgs() command line template.
  * Added Workbench interface.
  * Added preferences settings, including a preferences tool (OLPrefs)
  * Added localisation support for strings.
  * Improved support for dlink linker.
  * See OL.doc and OLPrefs.doc for details.

OD

  * Completely new ReadArgs() command line template.
  * Added Workbench interface.
  * Added localisation support for strings.

OEL

  * Now reports the error number if it cannot find it in the catalog.
  * Added the ERRPATH argument to match OC.

OBumpRev

  * Changed the code generated to remove the need to import Exec.
  * Added the DATEONLY argument to update the date without changing the
    revision.

FPE

  * No longer uses arp.library.
  * Format of settings files has changed.

Preferences

  Preferences settings have been added to OC and OL, and preferences
  tools have been created for them (OCPrefs and OLPrefs respectively).
  These tools are very rudimentary (thanks Helmuth ;-) and will be
  improved with GUIs as soon as possible.

Localisation

  Localisation support has been added to the main programs: OC, OL, OD,
  OCPrefs and OLPrefs. OEL and OBumpRev already had it. The source
  description (.sd), catalog description (.cd) and catalog translation
```

(.ct) files for FlexCat 1.5 are in the OBERON-A:Catalogs directory. Catalog translation files and catalogs for German are in the OBERON-A:Catalogs/Deutsch directory, and for Italian in the OBERON-A:Catalogs/Italiano directory. The catalogs have been produced from the translation files using FlexCat 1.5, but have not been tested so far. Thanks to Helmuth Ritzer and Edmondo Tommasina for the translations.

Locations

Most of the main programs have now been moved out of the OBERON-A:C directory into the OBERON-A: directory. This was done so that they are easier to use from the Workbench, and can use a common Catalogs directory. Their old spots have been taken by a small utility that will report any attempt to call them using their old paths.

Environment settings

A new env-archive directory containing default icons has been added. This should be copied to ENV: during the startup sequence.

Module SYSTEM

* Removed the following defunct procedures:
* SYSTEM.NEW no longer allows an option memory requirements parameter. Use Kernel.Allocate instead.
* Added the procedure CC.

Module Kernel

* Now fully integrated with the compiler as the run-time system.
* Procedure New renamed to Allocate. Procedure NewFromTag renamed to New.

Amiga interface modules

* Constant names have been modified to match those in the AmigaOberon interfaces. This mainly involves deleting any type prefix, except where this would cause identifier clashes. For example, idcmpCloseWindow becomes closeWindow.

* Type extensions are now made indirectly, through dummy base types, with the former base type becoming the first field in the new type.

  For example, the old interfaces treated extensions like this:

```
TYPE
  Node = RECORD
    name : LSTRPTR;
    ...
  END;
  Message = RECORD (Node)
    ...
  END;
```

  For a variable of type Message, the name field was accessed like this:

```
  name := message.name;
```

The new interfaces treat the same declarations like this:

```
TYPE
  NodeBase = RECORD END;
  Node = RECORD (NodeBase)
    name : LSTRPTR;
    ...
  END;
  MessageBase = RECORD (NodeBase) END;
  Message = RECORD (MessageBase)
    node : Node;
    ...
  END;
```

For a variable of type Message, the name field is now accessed like this:

```
  name := message.node.name;
```

It is now likely that type casts using SYSTEM.VAL will be required more frequently.

* Library interface modules no longer export an OpenLib() procedure. Instead, they always attempt to open the library in their initialisation code and it is up to the programmer to check for success.

* The following modules have been added, to match those in the AmigaOberon interfaces:

  - ExecSupport
  - ClassFace
  - RVI

* Exec - the LSTRPTR type has been added and the STRPTR type changed to conform with AmigaOberon. Interface modules have now been changed to use LSTRPTR instead of STRPTR. The constants NILSTR and EMPTYSTR have been added. The BSET and WSET types have been deleted, and the types Sets.SET8 and Sets.SET16 should be used instead.

Amiga utility modules

* Args - now obsolete, moved to OBERON-A:Source/Obsolete. Use Dos.ReadArgs instead.
* BoopsiUtil - now obsolete, moved to OBERON-A:Source/Obsolete. Use module ClassFace instead.
* DosUtil - added ObjectExists, DirExists and HaltIfBreak.
* ExecUtil - deleted the following procedures, which are now in module ExecSupport: NewList, CreatePort, DeletePort, BeginIO, AbortIO, CreateExtIO, DeleteExtIO, CreateStdIO, DeleteStdIO, CreateTask and DeleteTask.
* HookUtil - now obsolete, moved to OBERON-A:Source/Obsolete. Procedures HookEntry and InitHook moved to module Utility.

```
  * RexxUtil - now obsolete, moved to OBERON-A:Source/Obsolete. Use
    module RVI instead.

Framework modules

  * Events - most methods are now implemented as type-bound procedures,
    improving the object-oriented style of the module.
  * GTEvents - now obsolete, moved to OBERON-A:Source/Obsolete. The
    types and procedures have now been moved to module Events.

Library modules

  * New modules: XYPlane, In, Out, Out2, Math, MathL, Conversions,
    Sets and OberonLib.
  * Files - moved from OBERON-A:Source/ProjectOberon and
    re-implemented. Added procedures for machine independant formatted
    IO.
  * Strings - extensive changes to the interface to match the Oakwood
    recommendations. Procedures not in the Oakwood module moved to
    module Strings2.
  * Lists - interface redesigned and improved.
  * Errors - reporting of errors improved. Now requires to be
    initialised by calling Errors.Init.
  * BigSets - now obsolete, moved to OBERON-A:Source/Obsolete. Use
    module Sets instead.
  * StdIO - now obsolete, moved to OBERON-A:Source/Obsolete. Use
    modules In and Out instead.

Project Oberon modules

  * Files: moved to OBERON-A:Source/Library and re-implemented.
```

1.11 Release 1.5 update 1

```
Bugs fixed
----------

Module Files
  * A call to Dos.SetFileSize caused Enforcer hits on at least one
    system. There seemed no reason for it, but the statement was
    unnecessary, so it was removed. This bug affected all programs
    using module Files, including OC and FPE.

FPE
  * The lock on the current directory was being released too soon.

OD
  * Didn't properly handle non-exported type-bound procedures.

Changes made
------------

Module Oberon
  * Renamed to OberonClock to avoid a name clash with a module in a
    partial implementation of the Oberon System.
```

```
Module AmigaSupport
  * New module. Provides a screen and window to be used by a partial
    implementation of the Oberon System.

Module InputPO
  * New module. Implements the same interface as the Oberon System
    Input module.

Module XYPlane
  * Changed to use AmigaSupport and InputPO.

Module Files
  * Changed to use OberonClock instead of Oberon.

OC
  * Changed to use OberonClock instead of Oberon.

Other programs
  * Recompiled and relinked to use fixed Files module.
```

1.12 Release 1.5 update 2

```
Bugs fixed
----------

OC
  * Didn't properly handle code buffer overflows.
  * Comparing a value ARRAY OF CHAR procedure parameter with an empty
    string generated invalid code.
  * Multiplication of SHORTINTs was completely broken.

Improvements made
-----------------

OC
  * Overflow checks after multiplications now implemented.
  * Range checks now implemented for calls to the standard procedure
    SHORT.

Module Errors
  * Now reports multiplication overflows and range checks.
```

1.13 Release 1.5 update 3

```
Bugs fixed
----------

Oberon-A.sd

  * The source description file for FlexCat contained a bug that was
    accordingly introduced into all localised programs. This caused
```

unexplained long pauses and Enforcer hits.

1.14 Release 1.6

```
Introduction
------------

Release 1.6 represents a progressive development of Oberon-A following
the major restructuring of Release 1.5. The most significant changes
are in the compiler's code generation, and in the preferences editors.

Bugs fixed
----------

OC

  * If a unexported type-bound procedure was declared for a type in one
    module, and an extension in another module had a type-bound
    procedure of the same name, the compiler crashed with a NIL check
    trap when translating a call to that procedure.
  * Passing a string constant to a VAR ARRAY OF CHAR parameter caused
    the compiler to crash.

FPE

  * An Enforcer hit occurred if FPE was run from the Shell with no
    arguments.

Changes made
------------

OC

  * Improved the code generated by remembering the contents of address
    registers in some circumstances.
  * Implemented the Small code, Small data and Resident data models.
  * The size of the code and constant buffers can now be set in
    preferences, and can exceed 32K.
  * Floating point constant expressions are now allowed.
  * Greatly simplified the command line template by removing arguments
    that over-rode preferences settings.
  * Increased the maximum depth of type extensions from 7 to 15.
  * Implemented the AssertChk pragma.
  * Implemented the REGISTER option.

OCPrefs

  * Implemented a completely new font-adaptive GUI using EAGUI.library.

OL

  * Disabled the support for DLink.

OLPrefs
```

```
  * Implemented a completely new font-adaptive GUI using EAGUI.library.

OD

  * Added options to generate a compilable skeleton module (contributed
    by hartmut Goebel).

Localisation

  * Module Errors is now localised.
  * Updated the Italian and German catalogs (thanks again to Edmondo
    Tommasina and Helmuth Ritzer).

Module Kernel

  * Now supports registration of modules, types and commands.
  * Memory lists are now guarded by semaphores to support
    multi-threading. More work needs to be done in this area.

Framework modules

  * Module Events now allows for the garbage collector to be called
    regularly as part of the event loop.

Amiga interface modules

  * Utility.InitHook() now supports all data models.

Examples

  * Implemented the Oberon0 System.
```