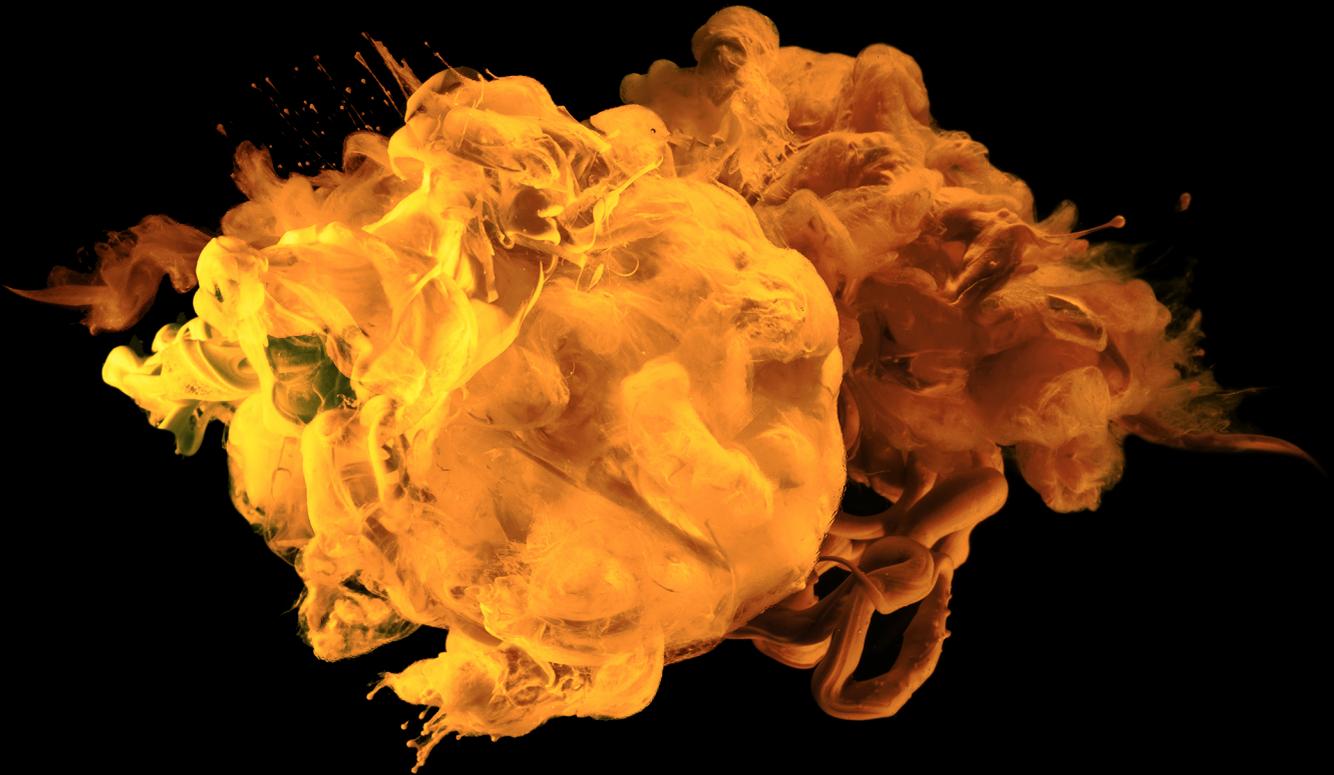


Dyalog for macOS Installation and Configuration Guide

Dyalog version **18.0**



DYALOG

The tool of thought for software solutions

*Dyalog is a trademark of Dyalog Limited
Copyright © 1982-2020 by Dyalog Limited
All rights reserved.*

Dyalog for macOS Installation and Configuration Guide

Dyalog version 18.0
Document Revision: 20200603_180

Unless stated otherwise, all examples in this document assume that □IO □ML ← 1

No part of this publication may be reproduced in any form by any means without the prior written permission of Dyalog Limited.

Dyalog Limited makes no representations or warranties with respect to the contents hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. Dyalog Limited reserves the right to revise this publication without notification.

*email: support@dyalog.com
<https://www.dyalog.com>*

TRADEMARKS:

SQAPL is copyright of Insight Systems ApS.

Array Editor is copyright of davidliebtag.com

Raspberry Pi is a trademark of the Raspberry Pi Foundation.

Oracle®, Javascript™ and Java™ are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

Linux® is the registered trademark of Linus Torvalds in the U.S. and other countries.

Windows® is a registered trademark of Microsoft Corporation in the United States and other countries.

macOS® and OS X® (operating system software) are trademarks of Apple Inc., registered in the U.S. and other countries.

All other trademarks and copyrights are acknowledged.

Contents

1	About This Document	1
1.1	Audience	1
1.2	Conventions	1
2	Introduction	3
3	Pre-requisites for Installation	4
3.1	Hardware Requirements	4
3.2	Operating System Requirements	4
3.3	Hard Drive Requirements	5
4	Installing/Upgrading Dyalog	6
5	Verifying a Successful Installation	7
6	Configuration	8
6.1	Setting the Dyalog Serial Number	9
6.2	Customising the Dyalog Session	10
6.3	Installing the APL385 Unicode Font	10
6.4	Installing New Keyboard Key Mappings	11
7	Uninstalling Dyalog	14
A	Default Keyboard	15
B	Configuration Files	16
B.1	Configuration File Structure	17
B.2	Arrays	18
B.3	References to other Configuration Parameters	18
B.4	Nested Structures	18
B.5	Example Configuration File Content	18
C	Configuration Parameters	20
	Index	24

1 About This Document

This document covers the installation procedure and configuration options for Dyalog on macOS. It details the pre-requisites for installation and the configuration parameters that can be used to customise individual instances.

This document is only part of the full documentation set for Dyalog (updated regularly and available to download from <https://docs.dyalog.com>). In cases where the information given is different between this document and any of the others, this document should be regarded as the definitive source for Dyalog for macOS.

 A complete documentation set is installed in the **[DYALOG]/help** directory when Dyalog is installed. The path to the **[DYALOG]** directory (that is, the directory in which Dyalog is installed) can be found from within a Session by entering:

```
+2␣NQ ' .' 'GetEnvironment' 'DYALOG'
```

1.1 Audience

It is assumed that the reader has a basic knowledge of macOS and software terminology; no prior knowledge of Dyalog is required.

1.2 Conventions

Unless explicitly stated otherwise, all examples in Dyalog documentation assume that `␣IO` and `␣ML` are both 1.

Various icons are used in this document to emphasise specific material.

General note icons, and the type of material that they are used to emphasise, include:

 Hints, tips, best practice and recommendations from Dyalog Ltd.

-  Information note highlighting material of particular significance or relevance.
-  Legacy information pertaining to behaviour in earlier releases of Dyalog or to functionality that still exists but has been superseded and is no longer recommended.
-  Warnings about actions that can impact the behaviour of Dyalog or have unforeseen consequences.

Although the Dyalog programming language is identical on all platforms, differences do exist in the way some functionality is implemented and in the tools and interfaces that are available. A full list of the platforms on which Dyalog version 18.0 is supported is available at <https://www.dyalog.com/dyalog/current-platforms.htm>. Within this document, differences in behaviour between operating systems are identified with the following icons (representing macOS, Linux, UNIX and Microsoft Windows respectively):



2 Introduction

Dyalog for macOS is a complete 64-bit Unicode implementation of Dyalog. It contains the same APL language engine as all other variants of Dyalog and is compatible with Dyalog running on other supported platforms – workspaces and component files can be copied between platforms or shared in real time on a network without needing to undergo conversion.

The development environment on the Mac is the Remote Integrated Development Environment (RIDE). The default Mac installation causes RIDE and the APL Engine to start simultaneously and appear to be a single integrated unit. However, the RIDE can also be used to connect to remote Dyalog engines running on any platform. For more information about the full functionality of RIDE as a remote development environment, see the *RIDE User Guide*.

3 Pre-requisites for Installation

This chapter details the conditions that must be met before installing Dyalog on the macOS operating system.

3.1 Hardware Requirements

The target Mac must have been introduced in 2010 or later.

To check the year of introduction of your Mac

1. In the Apple (🍏) menu in the menu bar, select **About This Mac**.
A window detailing the technical specifications is displayed.
2. Open the **Overview** tab.
An overview of your Mac is displayed, including its model and when it was introduced, for example, "Mac mini (Early 2011)" or "MacBook Pro (Mid 2014)".

3.2 Operating System Requirements

Dyalog version 18.0 is supported on macOS High Sierra onwards.

To check the operating system on your Mac

1. In the Apple (🍏) menu in the menu bar, select **About This Mac**.
A window detailing the technical specifications is displayed.
 2. Open the **Overview** tab.
An overview of your Mac is displayed, including its operating system.
-

3.3 Hard Drive Requirements

There should be at least 0.5 GB of free capacity on the target Mac.

To check the free capacity of your hard drive

1. In the Apple (🍏) menu in the menu bar, select **About This Mac**.
A window detailing the technical specifications is displayed.
 2. Open the **Storage** tab.
The storage information for your hard drive is displayed.
-

4 Installing/Upgrading Dyalog

This chapter is concerned both with new installations of Dyalog and with upgrading existing installations of Dyalog.

To install/upgrade Dyalog

1. Download **mac_64_<version>_unicode.pkg** from my.dyalog.com.
 2. Double-click on the **mac_64_<version>_unicode.pkg** file.
The **Dyalog Installer** window is displayed.
 3. Follow the instructions in the **Dyalog Installer** window.
 4. Close the **Dyalog Installer** window.
-

Dyalog is now installed and ready to use.

5 Verifying a Successful Installation

To verify that Dyalog has been successfully installed

1. Open **Finder**.
2. In the **Go** menu in the **Finder** menu bar, select **Applications**.
The **Applications** window is displayed.
3. In the **Applications** window, double-click on **Dyalog-<version>**.
A Dyalog Session is started.

Starting a Dyalog Session adds the Dyalog icon to the dock. To keep the Dyalog icon in the dock permanently, right-click on the icon and select **Options > Keep in Dock** from the drop-down list that appears.

6 Configuration

Although no configuration is necessary and Dyalog can be used as soon as it has been installed, various configuration options are available so that Dyalog can be customised to meet personal preferences.

During a Dyalog Session, the **Preferences** dialog box can be accessed from the **Session** window. For details of the configurations available through the **Preferences** dialog box, see the *Dyalog for macOS UI Guide*.

In addition to configuring Dyalog, the APL385 font can be installed along with a locale-specific keyboard. This enables Dyalog glyphs to be entered in other applications (for example, Mail and TextEdit).



Some configuration options require changes to files in the **\$HOME/.dyalog** directory. This is a "hidden" directory; to display it so that you can access the files it contains you need to toggle the display of "hidden" directories.

To toggle display of "hidden" directories

1. Open **Finder**.
 2. In the **Go** menu in the **Finder** menu bar, select **Go to Folder...**
A dialog box is displayed.
 3. In the field in the dialog box, enter a directory name of **Users/<username>/**.
The listing for the **Users/<username>/** directory is displayed.
 4. Press the following key combination: **⌘ + Shift + .**
The display of "hidden" files is toggled on.
-

6.1 Setting the Dyalog Serial Number

If you have registered your copy of Dyalog or have a commercial licence then you will have been sent a Dyalog serial number; this serial number is individual to you and corresponds to the type of licence that you are entitled to use.

Dyalog Ltd recommends setting the serial number either by editing a file containing the serial number directly or by running a function in a Dyalog Session to update the file containing the serial number. The next time Dyalog is started after setting the serial number, the `DYALOG_SERIAL` environment variable is set to the contents of this file.



If the `DYALOG_SERIAL` environment variable already exists and has a non-empty value, then its value is not updated with the contents of the serial file.

In a multi-user environment it might be desirable to set the `DYALOG_SERIAL` environment variable in a system configuration file so that the serial number is held in a single location.

To set your Dyalog serial number by editing the serial number file directly

1. In your preferred text editor, edit the `$HOME/.dyalog/serial` text file so that it contains:

```
serialnumber
```

where *serialnumber* is your Dyalog serial number.

2. Save `$HOME/.dyalog/serial`.



`$HOME/.dyalog/serial` is the default location for your serial number file but you can set the `DYALOG_SERIALFILE` environment variable to point to any other valid location.

To set your Dyalog serial number from within a Session

1. In a Dyalog Session, enter:

```
[SE.Dyalog.Serial serialnumber
```

where *serialnumber* is your Dyalog serial number.

The Dyalog serial number is saved in the serial number file `$HOME/.dyalog/serial`.

2. Exit and restart the Session.
-

Your serial number is displayed in the banner when you start a Session. To see your serial number at any time, enter:

```
+2[NQ].' 'GetEnvironment' 'DIALOG_SERIAL'
```

or

```
[SE.Dyalog.Serial ''
```



Using or entering a serial number other than the one issued to you is not permitted. Transferring the serial number to anyone else is not permitted. For the full licence terms and conditions, see

https://www.dyalog.com/uploads/documents/terms_and_conditions.pdf.

6.2 Customising the Dyalog Session

Dyalog can be customised using configuration parameters. These can be set in various ways; if a configuration parameter is set in multiple places the following descending order of precedence applies:

1. command line settings
2. application configuration file settings
3. environment variable settings
4. user configuration file settings
5. built-in defaults

This provides a great deal of flexibility, enabling a user to override one setting with another. For example, a "usual" workspace size (MAXWS) can be defined in the user configuration file, but be temporarily superseded by entering a different value when starting a Dyalog Session from the command line.

For more information on configuration files, see *Appendix B*. For a complete list of the configuration parameters that can be used to configure Dyalog, See *Appendix C*.

6.3 Installing the APL385 Unicode Font

When Dyalog is installed, the APL385 font that is used to display the APL glyphs correctly is embedded in the Session. To use this font in other applications (for example, email) it needs to be installed.

To install the APL385 Unicode font

1. Download the APL385 Unicode font from <https://www.dyalog.com/apl-font-keyboard.htm>.
 2. Open **Finder**.
 3. In the **Go** menu in the **Finder** menu bar, select **Computer**.
The **System** directory is displayed.
 4. Double-click your hard disk icon.
The **Disk** directory is displayed.
 5. Double-click the Library icon.
The **Library** directory is displayed.
 6. Double-click the Fonts icon.
The **Fonts** directory is displayed.
 7. Copy the **Apl385.ttf** file into the **Fonts** directory (this requires administration rights).
 8. Exit the **Fonts** directory.
-

6.4 Installing New Keyboard Key Mappings

The default keyboard that is installed with Dyalog for use in a Session is shown in *Appendix A*. For information on customising this keyboard, see the *Dyalog for macOS UI Guide*.

To completely replace the default keyboard with a locale-specific keyboard in the Session or to enter Dyalog glyphs in other applications (for example, email), the requisite keyboard key mappings need to be installed and enabled.

To install the keyboard key mappings for Dyalog glyphs

1. Download and unzip the **DyalogAlt.zip** file specific to your locale from <https://www.dyalog.com/apl-font-keyboard.htm>.
This zip file contains the **DyalogAlt.keylayout** and **DyalogAlt.icns** files specific to the selected locale. A separate locale-specific **mac_keyboard.pdf** file can also be downloaded from this location – this contains images of the key mappings enabled with the **DyalogAlt.keylayout** and **DyalogAlt.icns** files.



If your locale does not have files on this webpage, please contact support@dyalog.com.

2. Open **Finder**.
 3. In the **Go** menu in the **Finder** menu bar, select **Computer**.
The **System** directory is displayed.
 4. Double-click your hard disk icon.
The **Disk** directory is displayed.
 5. Double-click the Library icon.
The **Library** directory is displayed.
 6. Double-click the Keyboard Layouts icon.
The **Keyboard Layouts** directory is displayed.
 7. Copy the **DyalogAlt.keylayout** and **DyalogAlt.icns** files into the **Keyboard Layouts** directory (this requires administration rights).
-

To enable the keyboard key mappings for Dyalog glyphs

1. Open **System Preferences**.
The **System Preferences** window is displayed.
2. Double-click the **Keyboard** icon.
The **Keyboard** window is displayed.
3. Open the **Keyboard** tab.
4. Check **Show Keyboard & Character Viewers** in menu bar.
The current keyboard selection is displayed in the menu bar.
5. Open the Input Sources tab.
6. Click **+**.
The available keyboards to include are listed.
7. Select *Others* from the list on the left-hand side.
The list on the right-hand side is updated to show the available keyboards.
8. Select *Dyalog Alt <locale>* from the list on the right-hand side and click **Add**.
The list on the left-hand side is updated to show the enabled keyboard layouts.
9. Open the **Shortcuts** tab.

10. Select *Input Sources* from the list on the left-hand side.

The list on the right-hand side is updated to show the shortcuts that can be configured; these move between the enabled keyboards.

11. Set the shortcuts for the two options (**Select the previous input source** and **Select next source in Input menu**). If Spotlight is not enabled, then the default values of **⌘ + Space** and **⌘ + ⇧ + Space** can be used; if Spotlight is enabled then they must be changed. Setting these values provides a simple means of switching between all enabled keyboards.
-

You can now use the Dyalog keyboard in applications other than Dyalog whenever the Dyalog keyboard is selected in the menu bar.

7 Uninstalling Dyalog

To uninstall Dyalog

1. Open **Finder**.
2. In the **Go** menu in the **Finder** menu bar, select **Applications**.
The **Applications** window is displayed.
3. Drag the **Dyalog-<version>** icon from the **Applications** window into the Trash.
Dyalog is now uninstalled.



Although Dyalog has been uninstalled, some directories/files might remain in the **Home** directory.

A Default Keyboard

The keyboard key mappings shown in *Figure A-1* are enabled whenever a Dyalog Session is started.



Figure A-1: The default Session keyboard key mappings (UK keyboard)

To access the glyphs in the lower right quadrant, press ` followed by the appropriate key.

To access the glyphs in the upper right quadrant, press ` followed by the **SHIFT** key with the appropriate key.



To completely replace the default keyboard with a locale-specific keyboard in the Session or to enter Dyalog glyphs in other applications (for example, email), see *Section 6.4*.

B Configuration Files

A configuration file is a text file containing configuration parameters and values. It can cascade, that is, it can extend (inherit) configuration values from other configuration files, and supplement and/or override them. Configuration files use JSON5 (a superset of standard JSON) syntax and are portable across all systems supported by Dyalog.

The key benefits of defining configuration parameters using configuration files include:

- Configuration files are text-based. They are, therefore, easily managed along with the source code for an application, using industry standard tools for source code management and continuous integration.
- Application configuration files can be placed in application folders and define the configuration settings for a specific application.
- User configuration files provide settings that are the same for all applications. Typically, these files are used to configure the development environment.
- Interpreter configuration can be performed in the same way across all supported platforms.
- Dyalog can be launched from a text file that defines a function, namespace or class. If a configuration file exists with the same name as this file (but with a **.dcfg** extension), then Dyalog will detect this on launching and use the configuration parameter settings it defines.
- Configuration files are easy to read, and can be written directly or by using `JSON` (which supports JSON5).
- Both application and user configuration files can cascade, overriding settings defined in a more generic configuration file; this simplifies the configuration of components which share some configuration.



Dyalog Ltd recommends that configuration files are used for all run-time applications, and that the use of environment variables for this purpose is eliminated.

There are two different types of configuration file:

- *A user configuration file* – this defines configuration values for the current (possibly only) user of the system. The first time a new version of Dyalog is launched it

creates and initialises a user configuration file called **\$HOME/.dyalog/dyalog.<version-specific>.dcfg**, where the version-specific information comprises the version number, edition and width. For example, a 64-bit Unicode edition of Dyalog version 18.0 will be identified as **180U64**. The name of this file should not be changed.

- An *application configuration file* – this contains configuration values associated with a specific application. This is created by the user and should be saved at the same level as the application. It can either be given the same name as the workspace/script that is loaded when the application starts (but with the extension **.dcfg**) or the name should be stored in the CONFIGFILE parameter.

An additional configuration file called **\$HOME/.dyalog/dyalog.dcfg** is also created the first time any version of Dyalog is run. This can be edited to include configuration parameter values that should always be applied irrespective of Dyalog version so that they do not have to be redefined in multiple version-specific user configuration files.



Prior to Dyalog version 18.0, configuration parameters could be specified as environment variables and set in the **\$HOME/.dyalog/dyalog.config** script. This is no longer referenced, and any settings that should be retained must be re-entered in the appropriate **\$HOME/.dyalog/dyalog.<version-specific>.dcfg** configuration file.

B.1 Configuration File Structure

Configuration files define configuration parameters using JSON5. A JSON object contains data in the form of key/value pairs and other JSON objects. The keys are strings and the values are the JSON types. A key and its value are separated by a colon (:) character. Entries (key/value pairs) are separated by comma (,) characters.

The top-level object defines an optional key called `Extend` and an optional object called `Settings`:

- `Extend` is a string value containing the name of a configuration file to import. The extended (imported) file can extend another configuration file. Configuration values from the imported file(s) can be overridden by redefining them. The file name is implicitly relative to the name of the file that imports it (any file name extension must be explicitly specified).
- `Settings` is an object containing the names of configuration parameters and their values. The values can be a string, a number or an array of strings.

The names and values correspond to configuration parameters, and names are not case sensitive. Any named values can be defined; an APL application could query the values using `+2⌈NQ'.' 'GetEnvironment' <name>` or using the `]Config` user command.

If the same name is defined multiple times within a configuration file then the first definition will be used and a warning will be generated.

B.2 Arrays

An array can be used to define file paths, for example, `WSPATH: ["/dir1", "/dir2"]`. The only parameters which can be defined as arrays are `WSPATH`, `WSEXT` and `CFEXT`.

B.3 References to other Configuration Parameters

Configuration parameters that are string values can include references to other configuration parameters (irrespective of where they are defined) using square bracket delimiters. For example, `MySetting: "[DYALOG]/MyFile"` will replace `[DYALOG]` with the value of the `DYALOG` configuration parameter.

If the referenced configuration parameter is not defined then no substitution will take place; the reference, including the square bracket delimiters, will remain in place.

To include literal square brackets in a string, prefix them with a `\` character.

B.4 Nested Structures

Configuration files support nested parameter structures by defining an object that corresponds to the structure. For example:

```
Captions: {
    Session: "My Dyalog Session"
    Status: "My Status window"
}

+2 [NQ '.' 'GetEnvironment' 'Captions\Session'
My Dyalog Session
```

B.5 Example Configuration File Content

```
{
Extend: "my_default_configuration.dcfg",
Settings: {
    // maximum workspace
    MAXWS: "2GB",
    WSPATH: ["/dir1", "/dir2", ""],
    UserOption: 123,
    ROOTDIR: "/my/root/directory",
```

```
// references to other configuration parameters
FNAME: "[rootdir]/filename",
}
}
```

C Configuration Parameters

This appendix details the configuration parameters that can be used to customise Dyalog (for information on how to use them, see *Section 6.2*). If a configuration parameter is defined through an environment variable then its name must be specified entirely in upper case.

 When a configuration parameter defines a byte size, the value must consist of an integer optionally followed immediately by a single character denoting the units to be used. Valid values for units are K (KiB), M (MiB) and G (GiB) (for completeness, T (TiB), P (PiB) and E (EiB) can also be specified). If no units are specified, then KiB is assumed.

Table C-1: Configuration parameters that can be used to configure Dyalog on macOS

Configuration Parameter	Description
APL_TEXTINAPLCORE	Whether the "Interesting Information" section is included in aplcore files (for more information, see the <i>Dyalog for macOS UI Guide</i>). Valid values are: <ul style="list-style-type: none"> 0 – the "Interesting Information" section is not included 1 – the "Interesting Information" section is included Default = 1
DEFAULT_DIV	Value of □DIV in a clear workspace. Default = 0 (can be 0 or 1)
DEFAULT_IO	Value of □IO in a clear workspace. Default = 1 (can be 0 or 1)

Table C-1: Configuration parameters that can be used to configure Dyalog on macOS (continued)

Configuration Parameter	Description
DEFAULT_ML	Value of □ML in a clear workspace. Default = 1 (can be 0, 1, 2 or 3)
DEFAULT_PP	Value of □PP in a clear workspace. Default = 10 (must be integer in range 1 to 34)
DEFAULT_PW	Value of □PW in a clear workspace. Default = 79 (must be integer in range 42 to 32767)
DEFAULT_RTL	Value of □RTL in a clear workspace. Default = 0 (must be integer in range 0 to 32767)
DEFAULT_WX	Value of □WX in a clear workspace. Default = 3 (can be 0, 1 or 3)
DYALOG	The directory in which Dyalog is installed. This is referred to throughout the documentation as [DYALOG] .
DYALOG_SERIAL	Your Dyalog serial number. This must be set to the serial number issued to you. If not set, then the software is unregistered. For the full licence terms and conditions, see https://www.dyalog.com/uploads/documents/terms_and_conditions.pdf .
DYALOG_SERIALFILE	The full path to the serial text file containing your Dyalog serial number.
ENABLE_CEF	Whether the HTMLRenderer is enabled. Valid values are: <ul style="list-style-type: none"> • 0 – the HTMLRenderer is not enabled (necessary on macOS if RConnect is in use) • 1 – the HTMLRenderer is enabled Default = 1

Table C-1: Configuration parameters that can be used to configure Dyalog on macOS (continued)

Configuration Parameter	Description
ERRORONEXTERNAL EXCEPTION	<p>How the interpreter should behave when an exception occurs in a call to an external DLL. Valid values are:</p> <ul style="list-style-type: none"> • 0 – terminate with a system error • 1 – generate an EXTERNAL DLL EXCEPTION error (91) – this is trappable. <p>Default = 0</p> <p> This should only be used (set to 1) during development, not in production code.</p>
HISTORY_SIZE	<p>The size of the buffer (in KB) used to store previously-entered (input) lines in the Session.</p> <p>Default = 10</p>
INPUT_SIZE	<p>The size of the buffer (in KB) used to store marked lines (lines awaiting execution) in the Session.</p> <p>Default = 10</p>
LOG_FILE	<p>Full path and name of the log file in which the Session log is recorded.</p> <p>Default = \$HOME/.dyalog/default.dlf</p>
LOG_FILE_INUSE	<p>Whether the Session log is saved in a Session log file and loaded the next time a Session is started. Valid values are:</p> <ul style="list-style-type: none"> • 0 – the Session log is not saved • 1 – the Session log is saved in a Session log file and loaded the next time a Session is started <p>Default = 1</p>
LOG_SIZE	<p>The size of the Session log's buffer.</p> <p>Default = 800</p>
MAXWS	<p>The workspace size (amount of memory allocated to the workspace) when a new Dyalog Session is started. Must be an integer greater than 4 MiB.</p> <p>Default = 256M</p>

Table C-1: Configuration parameters that can be used to configure Dyalog on macOS (continued)

Configuration Parameter	Description
SESSION_FILE	Full path and name of the file containing the <code>□SE</code> object that is loaded when a new Dyalog Session is started. This file defines any functions/variables stored in the <code>□SE</code> namespace. Default = [DYALOG]/default.dse
TRACE_ON_ERROR	Whether the Trace window is automatically displayed when execution of a defined function is suspended with an error. Valid values are: <ul style="list-style-type: none"> • 0 – Do not display the Trace window when execution of a defined function is suspended with an error • 1 – Display the Trace window when execution of a defined function is suspended with an error Default = 1
UCMDCACHEFILE	Full path and name of the user command cache file (for more information, see the <i>User Commands User Guide</i>). Default = \$HOME/.dyalog /UserCommand20.cache
WSEXT	The workspace filename extension, defined as a colon separated string of one or more extensions, including any <code>.</code> character that separates the filename from the extension. Used by <code>WSPATH</code> and when saving a workspace that does not have an explicitly-specified extension. Default = :.dws:.DWS
WSPATH	The workspace path. A list of directories that are searched by the <code>)LOAD</code> , <code>)COPY</code> and <code>)LIB</code> system commands in the order specified. Multiple directories are separated by the <code>:</code> character.

Index

C	
Configuration	8
Configuration files	16
Configuration parameters	
(available)	20
Configuration parameters (setting)	10
Configuration files	16
Configuration parameters (available)	20
Configuration parameters (setting) ..	10
D	
Default keyboard	15
Dyalog serial number	9
I	
Installation	6
Installing the APL font	10
K	
Keyboard key mappings	11
P	
Pre-requisites for installation	4
S	
Serial number	9
U	
Uninstalling Dyalog	14
Upgrading Dyalog	6