CedarCatalog.tioga
   Doug Wyatt, July 20, 1987 5:04:02 pm PDT

Cedar Package Catalog

Abstract: This catalog is a list of interesting packages and tools. The catalog is automatically created from the collection of maintainer-supplied entries.

XEROX

Xerox Corporation
Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, California 94304

For Internal Xerox Use Only

Catalog Components

AMEvents: [Cedar]<Cedar7.0>Top>AMEvents.df

AMModel: [Cedar]<Cedar7.0>Top>AMModel.df

AMProcess: [Cedar]<Cedar7.0>Top>AMProcess.df

AMTypes: [Cedar]<Cedar7.0>Top>AMTypes.df

Documentation: RTTypesDoc.tioga

BasicCedar: [Cedar]<Cedar7.0>Top>BasicCedar.df

BasicFinger: [Cedar]<Cedar7.0>Top>BasicFinger.df

Created by: Carl Hauser

Maintained by: Carl Hauser <CHauser.pa>

Documentation: BasicFingerDoc.Tioga

Keywords: finger, database application, idle, users, RPC, server

Abstract: BasicFinger is a component of the Finger package for finding out information about other users and machines. It notices interesting events, such as login and logout, and uses RPC to a finger server to record them in the finger database. Users can query the database using the Finger package in CedarChest.

BasicLoadees: [Cedar]<Cedar7.0>Top>BasicLoadees.df

BasicLoadState: [Cedar]<Cedar7.0>Top>BasicLoadState.df

BasicPackages: [Cedar]<Cedar7.0>Top>BasicPackages.df

Created by: various artists

Maintained by: Russ Atkinson <Atkinson.pa>

Documentation: BasicPackagesDoc.tioga, HashTableDoc.tioga

Keywords: command registry, data structures, hash table, lists, queues, random numbers, REF, ROPE, symbol tables

Abstract: BasicPackages is a collection of useful small packages that is loaded in the default Cedar system. It exports the following interfaces: Commander, PriorityQueue, Random, RedBlackTree, RefTab, RopeFile, RopeList, SymTab. Sadly, the documentation for these interfaces is contained in the source code for the interfaces.

BasicTime: [Cedar]<Cedar7.0>Top>BasicTime.df

BCDStuff: [Cedar]<Cedar7.0>Top>BCDStuff.df

Binder: [Cedar]<Cedar7.0>Top>Binder.df

> **Documentation:** BinderDoc.tioga
>
> **Commands:** Bind. Binder

BootEssentials: [Cedar]<Cedar7.0>Top>BootEssentials.df

BootPackages: [Cedar]<Cedar7.0>Top>BootPackages.df

BTree: [Cedar]<Cedar7.0>Top>BTree.df

> **Documentation:** BTreeDoc.tioga

BTreeSimple: [Cedar]<Cedar7.0>Top>BTreeSimple.df

CedarRelease: [Cedar]<Cedar7.0>Top>CedarRelease.df

> **Documentation:** Changes.tioga. ReleaseMessage.tioga. ReleaseCoordinator.tioga.
> CedarCatalog.tioga. Cedar.depends. Cedar.dfIncludes

CommandTool: [Cedar]<Cedar7.0>Top>CommandTool.df

> **Documentation:** CommandToolDoc.tioga, CommandToolCommands.tioga.
> CommandToolStructureDoc.tioga
>
> **Commands:** SetProcessProperty. SetProperty. GetProperties. GetProcessProperties
>
> **Abstract:** This document is a user's guide for the CommandTool in Cedar 7.0.

CommDriver: [Cedar]<Cedar7.0>Top>CommDriver.df

Communication: [Cedar]<Cedar7.0>Top>Communication.df

> **Created by:** Hal Murray and Alan Demers
>
> **Maintained by:** Alan Demers <Demers.pa>
>
> **Documentation:** CommunicationDoc.tioga
>
> **Keywords:** address. ARPA. communication. network. protocol. PUP. socket. XNS
>
> **Abstract:** Communication.df contains the basic type declarations for Pup. XNS. and Arpa
> addresses; it contains no implementations.

Compiler: [Cedar]<Cedar7.0>Top>Compiler.df

> **Maintained by:** CedarSupport↑.pa
>
> **Documentation:** CompilerDoc.tioga. CompilerMessagesDoc.tioga
>
> **Keywords:** Cedar language. compiler. compiler errors. compute server. programming tools.
> Tioga documents. user profile
>
> **Commands:** Compile. Compiler. ComplexCompile. RCompile. RCompiler

Abstract: This document describes the Cedar compiler and how to use it within the Cedar programming environment.

ComputeServerUser: [Cedar]<Cedar7.0>Top>ComputeServerUser.df

CrRPC: [Cedar]<Cedar7.0>Top>CrRPC.df

Created by: Al Demers

Maintained by: Al Demers <Demers.pa>, Bill Jackson <BJackson.pa>

Keywords: communication, Courier, interoperability, RPC, Sirocco, XNS

Abstract: (transport-independent?) (intended primarily as target for Sirocco)

DebugTool: [Cedar]<Cedar7.0>Top>DebugTool.df

Created by: Several folks

Maintained by: CedarSupport↑.pa

Documentation: DebugToolDoc.tioga

Keywords: debugging, network tools, programming tools, processes, teledebugging

Abstract: This tool appears automatically in the Debugger world, and may be created in the normal Cedar environment with the "Debug" command. The tool can be set up to inspect the environment in which it is running, or to teledebug an instance of Cedar running on another machine. In the Debugger world, the tool can be used to look at the normal Cedar environment which has been stopped by outloading it to disk. The DebugTool allows you to inspect and modify the state of parallel processes, and to enumerate the set of loaded packages and modules.

DFCommands: [Cedar]<Cedar7.0>Top>DFCommands.df

Documentation: DFCommandsDoc.tioga

Keywords: DF file, programming tools, version management

Abstract: DFCommands provides a Commander-style interface to three common DF functions: BringOver, SModel, and VerifyDF. Various options are available through switches.

DFPackage: [Cedar]<Cedar7.0>Top>DFPackage.df

Disk: [Cedar]<Cedar7.0>Top>Disk.df

DumbTerminalSupport: [Cedar]<Cedar7.0>Top>DumbTerminalSupport.df

EditTool: [Cedar]<Cedar7.0>Top>EditTool.df

EssentialStyles: [Cedar]<Cedar7.0>Top>EssentialStyles.df

ExtraIago: [Cedar]<Cedar7.0>Top>ExtraIago.df

Documentation: ExtraIagoDoc.tioga

Commands: LocalIago

Faces: [Cedar]<Cedar7.0>Top>Faces.df

Created by: Lots of folks

Maintained by: CedarSupport↑.pa

Keywords: device interface, processor, disk, Ethernet, mouse, keyboard, display

Abstract: Faces are processor-independent interfaces to I/O devices such as the disk, ethernet, keyboard, mouse, and display. Faces.df contains just these interfaces. The corresponding processor-dependent implementations of the Faces (called Heads) are in separate packages, one for each machine.

File: [Cedar]<Cedar7.0>Top>File.df

Documentation: FileNotes.tioga, DiskFormats.tioga

FileStream: [Cedar]<Cedar7.0>Top>FileStream.df

Documentation: FileStreamDoc.tioga

FormatDisk: [Cedar]<Cedar7.0>Top>FormatDisk.df

FS: [Cedar]<Cedar7.0>Top>FS.df

Documentation: FSDoc.tioga

Germ: [Cedar]<Cedar7.0>Top>Germ.df

Documentation: Germination.Tioga

GrapevineUser: [Cedar]<Cedar7.0>Top>GrapevineUser.df

HeadsCommon: [Cedar]<Cedar7.0>Top>HeadsCommon.df

HeadsDLion: [Cedar]<Cedar7.0>Top>HeadsDLion.df

Documentation: HeadChanges.Tioga

HeadsDorado: [Cedar]<Cedar7.0>Top>HeadsDorado.df

Iago: [Cedar]<Cedar7.0>Top>Iago.df

Documentation: IagoDoc.tioga

Idle: [Cedar]<Cedar7.0>Top>Idle.df

Imager: [Cedar]<Cedar7.0>Top>Imager.df

> **Created by:** Michael Plass and Doug Wyatt
>
> **Maintained by:** The Imager Implementors <ImagerImplementors↑.pa>
>
> **Documentation:** ImagerDoc.tioga. ImagerConversionDoc.tioga
>
> **Keywords:** artwork. color. device independence. display, fonts. graphics. illustration. Imager. images. Interpress. printing. sampled images. scanned images
>
> **Abstract:** Package for device-independent image generation in Cedar.

Inscript: [Cedar]<Cedar7.0>Top>Inscript.df

> **Maintained by:** CedarSupport↑.pa
>
> **Documentation:** InscriptDoc.tioga
>
> **Keywords:** input events. mouse. keyboard
>
> **Abstract:** Inscript is an input event streaming package.

Installer: [Cedar]<Cedar7.0>Top>Installer.df

Interpreter: [Cedar]<Cedar7.0>Top>Interpreter.df

InterpreterTool: [Cedar]<Cedar7.0>Top>InterpreterTool.df

> **Maintained by:** CedarSupport↑.pa
>
> **Documentation:** InterpreterToolDoc.tioga
>
> **Keywords:** Cedar language. interpreter. debugging. programming tools. teledebugging
>
> **Abstract:** This document describes the InterpreterTool for Cedar. The InterpreterTool is a typescript-style tool with a read/eval/print control loop at the top level. like the CommandTool. It is used primarily for debugging and testing. but it is also used for determining and occasionally changing status variables in a running system. perhaps one a world-swap or teledebug away. It provides several related facilities: parsing, evaluation and printing of Cedar expressions in the context of the running system: management of breakpoints: and interactive control of uncaught signals, breakpoints and other exceptional conditions.

IO: [Cedar]<Cedar7.0>Top>IO.df

> **Created by:** Mark R. Brown
>
> **Maintained by:** CedarSupport↑.pa
>
> **Documentation:** IODoc.tioga. IOConversionDoc.tioga
>
> **Keywords:** Cedar language. conversions. input/output. ROPE. stream
>
> **Abstract:** The IO interface implements a byte stream abstraction with a variety of friendly

services.

Loader: [Cedar]<Cedar7.0>Top>Loader.df

LoaderDriver: [Cedar]<Cedar7.0>Top>LoaderDriver.df

LoadState: [Cedar]<Cedar7.0>Top>LoadState.df

MakeBoot: [Cedar]<Cedar7.0>Top>MakeBoot.df

> **Documentation:** MakeBootDoc.tioga
>
> **Commands:** MakeBoot

MBQueue: [Cedar]<Cedar7.0>Top>MBQueue.df

MesaRuntime: [Cedar]<Cedar7.0>Top>MesaRuntime.df

> **Created by:** A cast of dozens
>
> **Maintained by:** CedarSupport↑.pa
>
> **Documentation:** MesaRuntimeDoc.tioga
>
> **Keywords:** errors. Mesa language. PrincOps. process. signals. traps
>
> **Abstract:** MesaRuntime is the lowest layer of the Cedar system. Its interfaces describe the machine architecture and define some basic types and operations. It implements non-microcoded instructions, and the basic machinery for processes, signals, and traps.

MicrocodeDLion: [Cedar]<Cedar7.0>Top>MicrocodeDLion.df

MicrocodeDorado: [Cedar]<Cedar7.0>Top>MicrocodeDorado.df

Nucleus: [Cedar]<Cedar7.0>Top>Nucleus.df

OptionalHeadsDLion: [Cedar]<Cedar7.0>Top>OptionalHeadsDLion.df

PGS: [Cedar]<Cedar7.0>Top>PGS.df

> **Created by:** James Eve
>
> **Maintained by:** Doug Wyatt <Wyatt.pa>
>
> **Documentation:** PGSDoc.tioga
>
> **Keywords:** Cedar language. compiler. Mesa language. parsing
>
> **Commands:** PGS. TableCompiler
>
> **Abstract:** The parser generator system (PGS) is a Mesa program which takes a context free grammar specified in Backus-Naur form as input and to produce compacted binary tables are output which can be used in conjunction wth the Mesa parser.

PrintTV: [Cedar]<Cedar7.0>Top>PrintTV.df

ProcessProps: [Cedar]<Cedar7.0>Top>ProcessProps.df

Pup: [Cedar]<Cedar7.0>Top>Pup.df

ReadEvalPrint: [Cedar]<Cedar7.0>Top>ReadEvalPrint.df

Real: [Cedar]<Cedar7.0>Top>Real.df

    Documentation: FloatingPointDoc.tioga

Rollback: [Cedar]<Cedar7.0>Top>Rollback.df

Rope: [Cedar]<Cedar7.0>Top>Rope.df

    Created by: Russ Atkinson

    Maintained by: Russ Atkinson <Atkinson.pa>

    Documentation: RopeDoc.tioga

    Keywords: data structures. immutable. ROPE, string

    Abstract: ROPE is Cedar's standard "string" type: an immutable garbage-collected sequence of characters. The Rope interface provides a large set of useful operations on ropes, including rope concatenation. subrope extraction. and rope scanning. A client can provide his own specialized implementation of ROPE. The standard implementation of ROPE avoids copying for most rope concatenation and subrope extraction operations. Because they are immutable. ropes can be safely shared between programs without concern for storage ownership or synchronization.

Rosary: [Cedar]<Cedar7.0>Top>Rosary.df

    Documentation: RosaryDoc.tioga

    Keywords: data structures, lists. REF. ROPE. trees

    A client package that provides the analogy for ROPEs, with REF ANYS in place of CHARS.

RPCRuntime: [Cedar]<Cedar7.0>Top>RPCRuntime.df

SafeStorage: [Cedar]<Cedar7.0>Top>SafeStorage.df

    Documentation: SafeStorageDoc.tioga. SafeStoragePrimerDoc.tioga. SafeStoragePaper.tioga

SimpleTerminal: [Cedar]<Cedar7.0>Top>SimpleTerminal.df

STP: [Cedar]<Cedar7.0>Top>STP.df

SystemNames: [Cedar]<Cedar7.0>Top>SystemNames.df

Tapes: [Cedar]<Cedar7.0>Top>Tapes.df

Tentacles: [Cedar]<Cedar7.0>Top>Tentacles.df

Terminal: [Cedar]<Cedar7.0>Top>Terminal.df

TerminalCoordination: [Cedar]<Cedar7.0>Top>TerminalCoordination.df

ThisMachine: [Cedar]<Cedar7.0>Top>ThisMachine.df

Time: [Cedar]<Cedar7.0>Top>Time.df

Tioga: [Cedar]<Cedar7.0>Top>Tioga.df

> **Created by:** Bill Paxton
>
> **Maintained by:** TiogaImplementors↑.pa
>
> **Documentation:** TiogaDoc.tioga, TiogaAccessDoc.tioga, ViewersAndTiogaLocking.tioga, Status.tioga, TiogaConversion6.0.tioga
>
> **Keywords:** abbreviations, artwork, color, composition, conversion, definition search, Cedar interface, Cedar language, document model, documentation, editor, EditTool, formatting, page layout, printing, Tioga documents, TIP tables, typesetting, styles, user profile, version map, WYSIWYG
>
> **Abstract:** Tioga is a system to help you prepare documents. Its two main components are an editor and a typesetter. The editor lets you prepare the textual content of a document. The typesetter composes the document into pages for printing. Tioga is capable of dealing with simple technical papers and memos, and is well integrated within Cedar to support more mundane tasks such as writing programs. In future versions, it will be suitable for complex technical documents and books and will support tables, math formulas, and illustrations containing synthetic graphics and scanned images.
>
> When viewing this document on-line, use the level-clipping functions to see the overall structure rather than simply plowing straight through. LEFT-click the "Levels" button in the top menu, then LEFT-click "FirstLevelOnly" in the new menu that appears. This will show you the major section headings. LEFT-click "MoreLevels" to see the subsections, or "AllLevels" to read the details.

TIP: [Cedar]<Cedar7.0>Top>TIP.df

> **Created by:** Winiger & Paxton
>
> **Maintained by:** CedarSupport↑.pa
>
> **Documentation:** TIPDoc.tioga, TIPDocObsolete.tioga
>
> **Keywords:** input, keyboard, mouse, parsing, TIP tables, user interface
>
> **Abstract:** TIP (Terminal Interface Package) is a package built on top of Dan Swinehart's Inscript software that parses hardware-level actions from the keyboard, mouse, and keyset, into higher-level user actions. The input to TIP is a Mesa-like program (called here a TIP

table) describing the parsing algorithm. i.e. the set of events to be recognised, and the output to be passed through a client supplied procedure, which is called once per event parsed. TIP runs as a separate process from the client and is reentrant, allowing several instances to run at once. There are public interfaces to create a new TIP client and to change the parse table by which TIP recognises events. Users should be aware of the InputFocus interface that allows input events to arbitrarily directed to a single Cedar document.

TJaM: [Cedar]<Cedar7.0>Top>TJaM.df

TRope: [Cedar]<Cedar7.0>Top>TRope.df

TTY: [Cedar]<Cedar7.0>Top>TTY.df

UserCredentials: [Cedar]<Cedar7.0>Top>UserCredentials.df

UserProfile: [Cedar]<Cedar7.0>Top>UserProfile.df

> Maintained by: CedarSupport↑.pa
>
> Documentation: UserProfileDoc.tioga
>
> Abstract: Many components of Cedar permit the user to parameterize their behavior along certain predefined dimensions via a mechanism called the User Profile. Whenever the user boots or rollsback, his user profile is consulted to obtain the value for these parameters.

VersionMap: [Cedar]<Cedar7.0>Top>VersionMap.df

ViewerIO: [Cedar]<Cedar7.0>Top>ViewerIO.df

> Maintained by: CedarSupport↑.pa
>
> Keywords: typescript, stream, clunky user interfaces
>
> Abstract: A client package for creating a typescript viewer with associated input and output streams. Refer to comments in ViewerIO.mesa for further information.

Viewers: [Cedar]<Cedar7.0>Top>Viewers.df

> Documentation: ViewersDoc.tioga. ToolDesignDoc.tioga

VM: [Cedar]<Cedar7.0>Top>VM.df

> Documentation: VMDoc.tioga
>
> Keywords: memory allocation. page fault. performance. tuning, virtual memory
>
> This document describes VM. the interface to Cedar's virtual memory. It also includes some of the design decisions that influenced the VM implementation.

Watch: [Cedar]<Cedar7.0>Top>Watch.df

> Documentation: WatchDoc.tioga

Keywords: CPU usage, idle, memory allocation, network tools. page faults, performance measurement, power off, timing

Abstract: Watch is a program that maintains and displays statistics on selected system resources and events. Watch also has an automatic power off feature, which will power off the machine provided that there is no load on the machine.

WorldVM: [Cedar]<Cedar7.0>Top>WorldVM.df

XNSAuthentication: [Cedar]<Cedar7.0>Top>XNSAuthentication.df

Created by: Al Demers

Maintained by: Al Demers <Demers.pa>. Bill Jackson <BJackson.pa>

Keywords: authentication, communication, DES encryption, credentials, interoperability, network, password

Abstract: Use the XNSAuth interface.

XNSBasicTypes: [Cedar]<Cedar7.0>Top>XNSBasicTypes.df

XNSClearinghouse: [Cedar]<Cedar7.0>Top>XNSClearinghouse.df

XNSServerLocation: [Cedar]<Cedar7.0>Top>XNSServerLocation.df

XNSTransport: [Cedar]<Cedar7.0>Top>XNSTransport.df

Command Index

Bind: Binder
Binder: Binder
Compile: Compiler
Compiler: Compiler
ComplexCompile: Compiler
GetProcessProperties: CommandTool
GetProperties: CommandTool
LocalIago: ExtraIago
MakeBoot: MakeBoot
PGS: PGS
RCompile: Compiler
RCompiler: Compiler
SetProcessProperty: CommandTool
SetProperty: CommandTool
TableCompiler: PGS

Keyword Index

abbreviations: Tioga

address: Communication

ARPA: Communication

artwork: Imager, Tioga

authentication: XNSAuthentication

Cedar interface: Tioga

Cedar language: Compiler, InterpreterTool, IO, PGS, Tioga

clunky user interfaces: ViewerIO

color: Imager, Tioga

command registry: BasicPackages

communication: Communication, CrRPC, XNSAuthentication

compiler: Compiler, PGS

compiler errors: Compiler

composition: Tioga

compute server: Compiler

conversion: Tioga

conversions: IO

Courier: CrRPC

CPU usage: Watch

credentials: XNSAuthentication

data structures: BasicPackages, Rope, Rosary

database application: BasicFinger

debugging: DebugTool, InterpreterTool

definition search: Tioga

DES encryption: XNSAuthentication

device independence: Imager

device interface: Faces

DF file: DFCommands

disk: Faces

display: Faces, Imager

document model: Tioga

documentation: Tioga

editor: Tioga

EditTool: Tioga

errors: MesaRuntime

Ethernet: Faces

finger: BasicFinger

fonts: Imager

formatting: Tioga

graphics: Imager

hash table: BasicPackages

idle: BasicFinger, Watch

illustration: Imager
Imager: Imager
images: Imager
immutable: Rope
input: TIP
input events: Inscript
input/output: IO
interoperability: CrRPC, XNSAuthentication
Interpress: Imager
interpreter: InterpreterTool
keyboard: Faces, Inscript, TIP
lists: BasicPackages, Rosary
memory allocation: VM, Watch
Mesa language: MesaRuntime, PGS
mouse: Faces, Inscript, TIP
network: Communication, XNSAuthentication
network tools: DebugTool, Watch
page fault: VM
page faults: Watch
page layout: Tioga
parsing: PGS, TIP
password: XNSAuthentication
performance: VM
performance measurement: Watch
power off: Watch
PrincOps: MesaRuntime
printing: Imager, Tioga
process: MesaRuntime
processes: DebugTool
processor: Faces
programming tools: Compiler, DebugTool, DFCommands, InterpreterTool
protocol: Communication
PUP: Communication
queues: BasicPackages
random numbers: BasicPackages
REF: BasicPackages, Rosary
ROPE: BasicPackages, IO, Rope, Rosary
RPC: BasicFinger, CrRPC
sampled images: Imager
scanned images: Imager
server: BasicFinger
signals: MesaRuntime
Sirocco: CrRPC

socket: Communication
stream: IO, ViewerIO
string: Rope
styles: Tioga
symbol tables: BasicPackages
teledebugging: DebugTool, InterpreterTool
timing: Watch
Tioga documents: Compiler, Tioga
TIP tables: Tioga, TIP
traps: MesaRuntime
trees: Rosary
tuning: VM
typescript: ViewerIO
typesetting: Tioga
user interface: TIP
user profile: Compiler, Tioga
users: BasicFinger
version management: DFCommands
version map: Tioga
virtual memory: VM
WYSIWYG: Tioga
XNS: Communication, CrRPC