

ctys-vhost(1)

core address resolution interface
and database front-end

August 18, 2012

Contents

1	NAME	2
2	SYNTAX	2
3	DESCRIPTION	4
4	OPTIONS	6
	-c <args>	6
	MINCNT	6
	MAXCNT	6
	CNT	6
	-C <DB sources>	6
	ADJUST	7
	OFF	7
	MACMAPONLY	7
	MACMAP	7
	GROUPS	7
	LISTCACHE	8
	LISTTARGETS	8
	LISTGROUPS	8
	MEMBERSDB	8
	CLEARTMP	8
	CLEARALL	8
	REBUILDSCACHE	8
	-d <debug-level>	8
	-h	8
	-H <help-option>	8
	-i <input-list>	8
	CTYSADDRESS	8
	-I <0-9>	9
	0:OFF	9
	1:benchmark	9
	2:QUERY	9
	-l <login-name>	9
	-L <USER>	9
	-M <result-set-output-reduction>	9
	FIRST	9
	LAST	9
	ALL	9
	COMPLEMENT	9
	SORT	9
	USORT	9

UNIQUE	9
-o <output-list>	9
ACCELERATOR	11
ARCH	11
CATEGORY CAT	11
CONTEXTSTRING CSTRG	11
CTYSADDRESS CTYS	11
CTYSRELEASE	11
DIST	11
DISTREL	11
DNS	11
EXECLOCATION	11
EXEPATH	11
GATEWAY	11
GATEWAY	11
HWCAP	11
HWREQ	12
HYPERREL	12
HYPERRELRUN	12
IDS ID I	12
INDEX	12
IP	12
LABEL L	12
MAC M	12
MACHINE	12
MAXKEY	12
NETMASK	12
OS O	12
OSREL	12
PLATFORM PFORM	13
PM HOST H	13
PNAME P	13
REC_GEN[:<tab-args>]	13
RELAY	13
RELOCCAP	13
SERIALNUMBER SERNO	13
SERVERACCESS SPORT S	13
SPEC_GEN[:<tab-args>]	13
SSHPORT	13
SORT[:<sort-args>]	13
STACKCAP SCAP	13
STACKREQ SREQ	13
STYPE ST	13
TAB_GEN[:<tab-args>]	13
TCP T	14
TITLE	14
TITLEIDX	14
TITLEIDXASC	14
USERSTRING USTRG	14
UID	14
UUID U	14
VCPU	14
VERSION VERNO VER	14
VMSTATE VSTAT	14
VNCBASE	14
VNCDISPLAY DISP	14
VNCPORT CPORT	14
VRAM	14
XML_GEN[:<tab-args>]	14

-p <db-directory-path-list>	14
-r	15
-s <suppress-automatic-cache-rebuild>	15
-S <BasicDataManagementSupport>	15
LISTALL	15
LIST	15
LISTDB	15
MEMBERSDB	15
LISTGROUP	16
MEMBERSGROUP[:<group-list>]	16
MEMBERSGROUP2[:<group-list>]	16
MEMBERSGROUP3[:<group-list>]	16
MEMBERSGROUP4[:<group-list>]	16
MEMBERSGROUP5[:<group-list>]	16
MEMBERSGROUP6[:<group-list>]	16
MEMBERSGROUP6u[:<group-list>]	16
MEMBERSGROUP7[:<group-list>]	16
MEMBERSGROUP7u[:<group-list>]	16
MEMBERSGROUP8[:<group-list>]	16
MEMBERSGROUP8u[:<group-list>]	16
-T <type-list>	16
-V <shows-version>	16
-x <runtime states>	17
MARK	17
REVERSE R -	17
PING	17
SSH	17
PM	17
VM	17
-X <terse-output>	17
5 ARGUMENTS	17
6 EXIT-VALUES	22
7 SEE ALSO	23
8 AUTHOR	23
9 COPYRIGHT	23

1 NAME

ctys-vhost - core address resolution interface and database front-end

2 SYNTAX

ctys-vhost

```

[-c <spent cost on execution environment>
  =:(
    MINCNT|MAXCNT|CNT
  )
]
[-C <DB sources>
  =:(
    OFF
    |CLEARTMP
    |CLEARALL
    |GROUPS
    |KEEPALL
    |LIST
    |LISTCACHE
    |LISTTARGETS
    |LISTGROUPS
    |MEMBERSDB
    |MACMAPONLY
    |MACMAP
    |REBUILDCACHE
  )
]
[-d <debug-level>]
[-h]
[-H <help-options>]
[-i <input-list>=[CTYSADDRESS|CTYS]]
[-I <0-9>]
[-l <USER>]
[-M <result-set-output-reduction>
  =:(
    FIRST|LAST|ALL|COMPLEMENT|SORT|USORT|UNIQUE
  )
]
[-o <output-list>
  =:(
    (
      (
        [ARCH] [,]
        [ACCELERATOR|ACCEL] [,]
        [CATEGORY|CAT] [,]
        [CONTEXTSTRING|CSTRG] [,]
        [CPORT|VNCPORT] [,]
        [CTYSADDRESS|CTYS] [,]
        [CTYSRELEASE] [,]
        [DIST] [,]
        [DISTREL] [,]
        [EXECLOCATION] [,]
        [EXEPATH|EXEP] [,]

```

```

    [GATEWAY|GW] [,]
    [GROUPID|GID] [,]
    [HWCAP] [,]
    [HWREQ] [,]
    [HYPERREL|HYREL] [,]
    [HYPERRELRUN|HYRELRUN] [,]
    [IDS|ID|I] [,]
    [IFNAME] [,]
    [INDEX] [,]
    [LABEL|L] [,]
    [MAC|M] [,]
    [NETMASK] [,]
    [NETNAME] [,]
    [OS|O] [,]
    [OSREL] [,]
    [PLATFORM|PFORM] [,]
    [PM|HOST|H] [,]
    [PNAME|P] [,]
    [RELAY] [,]
    [RELOCCAP] [,]
    [SERIALNUMBER|SERNO] [,]
    [SERVERACCESS|SPORT|S] [,]
    [SSHPORT] [,]
    [STACKCAP|SCAP] [,]
    [STACKREQ|SREQ] [,]
    [TCP|T] [,]
    [TYPE|STYPE|ST] [,]
    [USERSTRING|USTRG] [,]
    [USERID|UID] [,]
    [UUID|U] [,]
    [VCPU] [,]
    [VERSION|VERNO|VER] [,]
    [VMSTATE|VSTAT] [,]
    [VNCBASE] [,]
    [VNCDISPLAY|DISP] [,]
    [VRAM] [,]
)
[TITLE|TITLEIDX|TITLEIDXASC] [,]
[MACHINE|MAXKEY] [,]
)
| (TAB_GEN|TAB) [:<tab-args>]
| (REC_GEN|REC) [:<tab-args>]
| (SPEC_GEN|SPEC) [:<tab-args>]
| (XML_GEN|XML) [:<tab-args>]
) [,]
[IP|DNS|D] [,]
[SORT[:ALL|A|UNIQUE|U|REVERSE|R|EACH
|[[[0-9][0-9][0-9]|0-9][0-9][0-9]|0-9]]]
[%]] [,]
]
[-p <db-directory-path-list>]
[-r]
[-s]
[-S <BasicDataManagementSupport>
=:(
    CONTENTGROUP
    |LISTALL
    |LIST
    |LISTDB

```

```

    | MEMBERSDB
    | LISTGROUP
    | MEMBERSGROUP([2345678])|([678]u)[:<groups-list>]
  )
]
[-T <type-list>]
[-V]
[-R <runtime states>
  =:(
    [MARK|(REVERSE|R|-),](PING|SSH)[,(PM|VM)]
  )
]
[-X]
<awk-regexp>[
  (
    AND
    |E:<#field0>:<#field1>
    |F:<#field>:<string-literal>
    |NOT
    |OR
    |<awk-regexp>
  )
]

```

3 DESCRIPTION

is the basic address resolution interface for runtime execution of commands based on ctys addressing ctys-vhost . The main task of this tool is to support a scripting interface for the functional link between VMs and PM, and their contained OS. Even though a considerable amount of functionality is provided, the user interface is designed in a manner for simplified application of the common daily tasks. The basic queries just require actually a few arguments, e.g.

```
ctys lab02 tst320
```

displays by default the record for the VM 'tst320' registered for execution on the PM 'lab03'. The following query

```
ctys lab02 tst32
```

displays on the PM 'lab02' all VMs with the pattern match '*tst32*', whereas

```
ctys lab02 tst3.0
```

displaye the pattern match '*tst3.0*' - where the '.' matches any character. When required a more detailed set of constraints on a query could be applied as required.

The 'ctys-vhost' utility combines functionality of:

Hostname resolution for physical and virtual machines, including off-line machines for a pre-start queries.

Address conversion into and from the extended syntax <machine-address> for network wide-unique addressing of VMs, PMs, and additionally HOSTs/Login-Sessions.

Dynamic availability checks including SSH-Logins by **ctys-vping**.

Inventory management front-end functions for various attributes generated by **ctys-vdbgen**.

Load balancing.

Integration of DHCP data from **ctys-extractMAClst**, **ctys-extractARPlst**, and **ctys-macmap**.

Import- and Export- functionality by various formats.

Handling of GROUPS objects in combination with **ctys-groups**.

Creation and refresh of cached pre-resolutions for network data.

Handling of multiple databases as specific **views**.

The basic similarity to the UNIX 'host' function for name resolution to virtual machines is expanded by several features, which take into account the roaming of VMs and thus changing their actual execution path within a so called 'execution stack' assembled by PMs, VMs, and HOSTs. This implies some dynamic data handling due to frequently changing 'physical containment positions' of roaming VMs. Therefore a basic load-balancing is included, which is currently still a pre-release.

Due to targeted simplicity and efficiency the address resolution is based on a combination of RECORD based and unstructured flat pattern matching by regular expressions only. Even though the record is structured by fields as given by '-a ENUMERATE=MACHINE', the match itself could be performed quite reliable and performant as a simple regex given by an '**awk-match**' string expression to be applied to RECORD-STRING, whereas the output is still displayed on the level of records and fields. Combined multiple-regexps are supported and are iteratively applied on the intermediary results of each step. The set of regular expression could be either combined by AND or OR operators, where AND is the default behaviour. This opens for more complex queries. The behaviour of the operator preference is described in detail within the following **ARGUMENTS** section. The filtering of the output as requested by the '-o' option is performed on the final set of resulting records only. This could be applied for example, when a table display has to be sorted, where the option '-o SORT:1' sorts the resulting table by the first column. The call with the option '-o sort:1,reverse' results in reverse sorting of the first column.

This simplified pattern matching approach is applicable when the given set of search attributes result in a unique match. When ambiguity occurs due to multiple matching records, the set of records are handled as defined by the '-M' option. Ambiguity occurs frequently when using a reduced set of selection attributes on a database for VMs in a networked environment. This is due to the frequently desired availability of multiple redundant access targets for the physically identical VM. Either the specification of additional attributes, or the application of load-distribution policies will resolve ambiguity. In current release a basic COST option '-C' extends the '-M' option for some usable load distribution within execution-groups.

In most other cases uniqueness should be given, e.g. the key UUID is defined to be applied as unique, but could be tampered unintentionally e.g. by co-allocated backups. For avoiding of backup-access the '-M first' option might be helpful, when the name of the backup is an extension. The more reliable approach here is to set the **VMSTATE** to **BACKUP** and using the **VMSTAT** filtering attribute for the ENUMERATE action when collecting the inventory data.

The current implementation enables the management of up to multiple thousands of VMs with the given simplicity with single-query responses in the range of about 0.6-0.8 seconds (measured, but ofcourse depends on machine etc.). One of the next versions will additionally support LDAP based nameservices, targeting an enterprise environment with a number of almost an unlimited amount of distributed services to be managed.

When a mapping table MAC-IPAddress-HostName - here named as 'macmap-DB' - is present this will be used for open mappings which are not configured within the VM configuration files. Particularly any IP or PM/Hostname address for a given MAC-Address is resolved when not present within the enum-DB.

The address resolution is performed by the following steps :

Check the static list of given ExecGroup for possible candidates.

List the active sessions on the given ExecGroup.

Take the appropriate PM/VM by utilising '-C' and/or '-M' option.

The following list shows some examples of using ctys-vhost as an interactive query tool.

- Given partial strings, e.g. '192.168.1' lists all machines of that subnet. When the '-M active' option is chosen, all currently active sessions within that subnet are listed.
- Any string could be used as partial pattern, e.g. parts of MAC-Ids or fragments of UUIDs. The given string will be matched against complete record, mostly an awk-regex, thus any part, even spanning multiple FIELDS could be used. But currently not regex, just literal characters are supported.
- The database founding the mapping information of ctys-vhost could be altered by '-p' option for handling of multiple sets, e.g. for test-purposes or reduced applied sets.

- The databases enum-DB and macmap-DB are populated just with the native information provided by their main sources, dhcpd.conf and the config-files of supported VMs. Therefore not any information might be present in each of them, e.g. the IP-Address of the GuestOS might be present within the macmap-DB, but not within the VM-config. The '-S' option allows for the selection or combined usage of multiple sources, e.g. by values 'all', 'macmap', or 'enum'.
- ctys specific configurations-extensions as described for the '-a ENUMERATE' option are fully supported. This includes particularly the storage of GuestOS information within the VM-config by specific ctys-Prefixes(#@#) and some helpful keywords.
The format of the generated data records is literally the same as the MACHINE output of the ENUMERATE action.

Additionally to the flat-matching by simple regular expressions some additional keywords are defined. These are AND , OR , NOT , E: ,and F: , described within the section related to the **ARGUMENTS**.

4 OPTIONS

-c <spent cost on execution environment>

Cost as for load distribution when selecting a target. Companion options apply to resulting set of equal cost.

<spent cost on execution environment>=(MINCNT|MAXCNT|CNT)

- **MINCNT:**
Gives minimum loaded target, number of given types are simply counted.
- **MAXCNT:**
Gives maximum loaded target, number of given types are simply counted.
- **CNT:**
Lists each target with it's TYPE-COUNT.

-C <DB sources>

Limits the generation of the cache DB to the for mapping-resolution to the listed sources. Default is to use all. Only available databases will be used, missing are silently ignored.

Due to some performance issues when repetitively accessing same temporary runtime data, some internal caches are defined. These can be controlled, and reused or cleared by usage of some of the following keywords. But additionally some automatic checks apply. For data from static information, which has to be pre-processed a local cache-DB is created. This cache-DB will be checked for modification time of it's sources before each access and updated when outdated. The modification time of the cache files will be checked additionally for their age. When these exceeds the value defined by CACHECLEARPERIOD, which is by default 3600seconds, the caches are forced-cleared and rebuild silently by next call.

The following data sources are utilized:

ENUM

Enumeration results only, as supplied by cached local"enum.fdb".

MACMAP

DHCP information for MAC resolution, the macmap-DB should be available, but is otherwise simply ignored. This will be utilized in conjunction with an enumeration result, e.g. ENUM.

GROUPS

Adds caching of GROUPS for all group files from the current CTYS_GROUPS_PATH variable. Therefore each group file will be completely expanded by nested evaluation and replacement of "#include" statements and stored by replacing each resulting entry with it's MACHINE format entry from the staticCacheDB.

Each group is cached within an file by it's own, thus the access could be performed by just one file-selection for the complete nested resolution of it's entities.

```
<DB sources>=
(
  OFF|
  CLEARTMP|
  CLEARALL|
  GROUPS|
  KEEPALL|
  LIST|
  LISTTARGETS|
  LISTGROUPS|
  MEMBERSDB|
  MACMAPONLY|
  MACMAP|
  REBUILDCACHE
)
```

This group of keywords controls the runtime behaviour, which has an impact to the overall performance.

- **ADJUST**
clears enum.fdb from redundant records of multiple scans. This is not necessarily required because the **REBUILDCACHE** clears redundant records before importing the current set.
- **OFF**
bypasses the usage of caches.
- **MACMAPONLY**
uses the macmap.fdb only for mapping, this is just senseful for mappings between DNS, MAC, and TCP. The request will be rejected, when "-o" option contains any other input.
For matching entities within MACMAP this might be the fastest approach. It is the only applicable approach, when the target is not yet populated in standard DB, for example due to pre-initial conditions.
- **MACMAP**
activates the raw usage of macmap.fdb for DNS, MAC, and TCP as preferred source of resolution.
This has two flavours, depending from selected output attributes:
 - Only one of, or all: TCP|MAC|DNS
In this case the MACMAP DB will be utilized within the "bigger awk", due complete probable containment of information thus first a raw access to MACMAP will be tried. When no result was found, the general script with DNS/Bind access will be performed. In standard manner(due to SW architecture, ignoring previous trial).
 - Additional output requested:
In this case particularly the field positions of the resulting output can not be handled in a smart manner for an independent pre-filter, though the standard execution path is performed.
When the macmap.fdb is properly maintained and contains the complete scope of mapping information, this enhances the performance, else it could have an negative impact, even though it will not be dramatic, or for small amounts almost not recognizable.
Another aspect to be aware of is, that the two different databases might diverge. Particularly the order of the stored records could not be relied on to be the same. When using the option "-M all" the order might not be relevant, but for "-M first"(default) and for "-M last" the results might frequently be different.
The basic difference of the contents is the fact, that the macmap.fdb (let us say!) contains any networked host, whereas the standard enum.fdb the registered VMs only, so might be a subset of macmap.fdb.
The correlation of both will be performed, when a cache is build and addressing references are resolved for faster access.

- **GROUPS**

activates the usage of GROUPS and it's related cache data which is due to performance issues deactivated for now by default.

The following additional keywords control and support the management of internal caches.

- **LISTCACHE**

lists all current caches. This call terminates immediately after performing, so any remaining options are ignored.

- **LISTTARGETS**

lists all current cached targets.

This call terminates immediately after performing, so any remaining options are ignored.

- **LISTGROUPS**

lists all current cached groups. This call terminates immediately after performing, so any remaining options are ignored.

- **MEMBERSDB**

displays a list of all current staticCacheDB members in ctys-stacked-address notation.

- **CLEARTMP**

clear it's internal temporary caches first and rebuild on demand.

- **CLEARALL**

clear all it's internal caches first and rebuild on demand. This includes a directory-wildcard-clear, which includes probably the caches of other tasks, so use it considerably.

This call terminates immediately after performing, so any remaining options are ignored.

- **REBUILDCACHE**

the static data to be concatenated from static assembly databases, for now the enum-DB and the macmap-DB is cached within a static database and concatenated with the volatile RT data into the RTCACHE. Redundant record from multiple-rescans are cleared.

The requirement of rebuild for the static data is checked by modification time of it's components, and when required updated silently. When setting this flag, the data is rebuild in any case.

Additional information is available from description of:

- NAMESERVICES of "ctys -H"
- "ctys -a ENUMERATE...."
- "ctys-extractMAClst"
- "ctys-vdbgen"

Same as ctys.

Print help, refer to '-H' for additional information.

-H <help-option>

The extended help option is based on system interfaces for display of manpages, PDF and HTML documents. This comprises the man pages and installed manuals.

For additional help refer to the documents or type **ctys -H help**.

-i <input-list>

Options controlling input content for specific cases.

`<input-list>=[CTYSADDRESS|CTYS]`

CTYSADDRESS|CTYS

A fully qualified address is supported for mapping of one of the given output attributes.

-I <0-9>

Interactive, gives summarised display of progress for main values. The degree of display depends on the chosen level:

- 0 For completeness only, switches the display OFF, same as omitting the option at all.
- 1 Activates a moderate level with display of basic benchmark data.
- 2 Activates a more informative level with intermediate QUERY data reduction pattern. This particularly supports the design of multi-key selection queries for performance optimization.

ctys-vhost <in-out-options> <arg1> <arg2> <agr3>

For the display of the actual contents of a specific intermediate step in addition to it's draft performance-overview, just drop all following filters/arguments from the call, what will display the requested result as final. This result is identical to the covered intermediate result when using it within a chained set of filters.

The users, which will be used for hosts without an explicitly given user. The hosts/groups entries provide the common EMail-Style "<user>@<execution-target>". The default is "\$USER", when neither "-l", nor an explicit user is provided.

-L <USER>

Remote user to be used for SSH-access-checks, when the "-x" option is activated.

DEFAULT=CTYS_NETACCOUNT(DEFAULT->USER)

-M <record-filter>

Restricts a set of multiple results with additional constrains for output:

<result-set-output-reduction>
=(FIRST|LAST|ALL|COMPLEMENT|SORT|USORT|UNIQUE)

- **FIRST**
First matching entity.
- **LAST**
Last matching entity.
- **ALL**
All matching entities.
- **COMPLEMENT**
All entities NOT matching.
- **SORT**
Final result is sorted by "sort".
- **USORT**
Final result is sorted by "sort -u". Only full matches are reduced.
- **UNIQUE**
Final result is sorted by "sort -u" but only displayed when actually one record only results. When multiple records are matched, an empty string is returned and the exit value is set to "1".

-o <output-list>

Options controlling output content. Values of all given options are listed as one RECORD per line for each matched entity. The keywords are not case sensitive and could be used as a comma-separated list. Shortcuts are applicable mostly as one-character alternatives as listed.

The default output when this option is not provided is to display a pre-configured table stored as a MACRO in the default-macros file with the name

TAB_CTYS_VHOST_DEFAULT

This table could be customized as required, but should be handled carefully.

```
<output-list>=
(
  (
    [ARCH] [,]
    [ACCELERATOR|ACCEL] [,]
    [CATEGORY|CAT] [,]
    [CONTEXTSTRING|CSTRG] [,]
    [CPORT|VNCPORT] [,]
    [CTYSADDRESS|CTYS] [,]
    [CTYSRELEASE] [,]
    [DIST] [,]
    [DISTREL] [,]
    [EXECLOCATION] [,]
    [EXEPATH|EXEP] [,]
    [GATEWAY|GW] [,]
    [GROUPID|GID] [,]
    [HWCAP] [,]
    [HWREQ] [,]
    [HYPERREL|HYREL] [,]
    [HYPERRELRUN|HYRELRUN] [,]
    [IDS|ID|I] [,]
    [IFNAME] [,]
    [INDEX] [,]
    [LABEL|L] [,]
    [MAC|M] [,]
    [NETMASK] [,]
    [NETNAME] [,]
    [OS|O] [,]
    [OSREL] [,]
    [PLATFORM|PFORM] [,]
    [PM|HOST|H] [,]
    [PNAME|P] [,]
    [RELAY] [,]
    [RELOCCAP] [,]
    [SERIALNUMBER|SERNO] [,]
    [SERVERACCESS|SPORT|S] [,]
    [SSHPORT] [,]
    [STACKCAP|SCAP] [,]
    [STACKREQ|SREQ] [,]
    [TCP|T] [,]
    [TYPE|STYPE|ST] [,]
    [USERSTRING|USTRG] [,]
    [USERID|UID] [,]
    [UUID|U] [,]
    [VCPU] [,]
    [VERSION|VERNO|VER] [,]
    [VMSTATE|VSTAT] [,]
    [VNCBASE] [,]
    [VNCDISPLAY|DISP] [,]
    [VRAM] [,]
  )
  [TITLE|TITLEIDX|TITLEIDXASC] [,]
  [MACHINE|MAXKEY] [,]
)
| TAB_GEN[:<tab-args>]
| REC_GEN|REC[:<tab-args>]
| SPEC_GEN|SPEC[:<tab-args>]
```

```

    | XML_GEN|XML[:<tab-args>]
)
[IP|DNS] [,]
[,SORT[:<sort-args>]]

```

The previous keywords for specific fields set the related bit for output. These will be OR-ed to the resulting output. Thus the MACHINE keyword includes all fields, whether individually set or not.

The format keys IP and DNS change the representation of the IP field.

- **ACCELERATOR**
The accelerator as configured.
- **ARCH**
The architecture presented by the hypervisor to the GuestOS.
- **CATEGORY|CAT**
The category of the plugin, which could be for now one of: HOSTs, PMs VMs.
- **CONTEXTSTRING|CSTRG**
A string stored for the use by responsible the plugin.
- **CTYSADDRESS|CTYS**
A fully qualified address to be used within ctys. This includes the complete address for the whole execution-stack of the destination instance, beginning with hosting PM.

Whereas almost any other output is just a subset of the generated static database, this value is the result of the assembly of multiple items to a complete address for an unambiguous execution path. The namespace could be the private network or even the global network, when globally unique PM addresses as FQDN are used.
- **CTYSRELEASE**
The release of ctys used for creation of the VM.
- **DIST**
Output of distribution installed within VMs guest.
- **DISTREL**
Release of distribution.
- **DNS**
Output of TCP/IP address (any valid for the VM). This option supports the name representation as reported by DNS, for the numerical representation refer to IP.
ATTENTION: Only the first match will be listed when multiple addresses are present for the same entity.
- **EXECLOCATION**
The location of execution for the VM. Either a keyword, or a list of hosts/groups.
- **EXEPATH**
The location of executable for starting the VM.
- **GATEWAY**
The TCP gateway to be used for the current interface, which is for the standard case the one for the whole multihomed node.
- **GROUPID**
The group id of user that created this entry.
- **HWCAP**
The offered hardware capacity by the VM to the GuestOS.

- **HWREQ**
The required hardware capacity of the VM from the PM, which could be a lower peer VM within a stack.
- **HYPERREL**
The release of the hypervisor the current VM is created with. E.g. "Xen-3.0-x86_64".
- **HYPERRELRUN**
The release of the present hypervisor when this record was created.
- **IDS|ID|I**
Output of static ID, which is a pathname for VMs, and a runtime ID for HOSTs. The IDs are (foreseen to be!?) unique within the namespace of their PM or VM. This should be considered when roaming VMs between PMs.
Following current exceptions apply:
 - **XEN**
The value is the configuration path statically unique on local host, common to IDs of other VMs.
The domain-ID is handled - due to hypervisor architecture and structural and dynamic means of accessibility - similar to an ordinary "UNIX-pid", but not considered within ctys.
 - **HOST**
For plugins of type HOST, which are more or less simple processes offering specific services, the "UNIX-ID" is utilized.
The "UNIX-ID" could consist of several kinds of entries. A common example is VNC, where the entries semantic could be one of:
 - * DISPLAY = VNC-port-offset
 - * DISPLAY = VNC-port
 - * Any of above could be context-specific, and utilized more or less correlated by any other FBP-aware application too. E.g. vncviewer for XEN and WMWare-Workstation 6.
In addition, for a plugin a ctys specific ID might be defined, e.g. based on "UNIX-PID".
So, ... it is just an abstract ID, no generic overall-algorithm applicable.
- **INDEX**
The index of the record within the current snapshot of the selected database. This is a transitive value, which may change for each database change.
- **IP**
Output of TCP/IP address. This option supports the numerical representation, for the DNS name representation refer to DNS.
- **LABEL|L**
Output of LABEL.
- **MAC|M**
Output of MAC address.
ATTENTION: Only the first match will be listed when multiple addresses are present for the same entity.
- **MACHINE**
Complete records matching the <regexpr-list> in terse format for postprocessing.
- **MAXKEY**
The maximum common set of attributes for LIST and ENUMERATE.
- **NETMASK**
The TCP netmask of current interface.
- **OS|O**
Output of OS as configured.
- **OSREL**
Release of OS.

- **PLATFORM|PFORM**
The HW platform provided for the GuestOS.
- **PM|HOST|H**
Output of TCP/IP address of the PM-Physical Machine, which is the hosting machine.
- **PNAME|P**
The same as <ID|I>, this is due to the usage of filepathname of the configuration as an unique ID at least within the namespace of a sigle hosts filesystem.
- **REC_GEN|REC**
Generates output format as structured proprietary record format, which is foreseen for online validation as well as automated postprocessing. Refer to common format of generic tables for additional information.
- **RELAY**
The relay interface, device, virtual bridge, virtual switch, or virtual hub, the VM is interconnected too witin it's PM/lower-stack-peer.
- **RELOCCAP**
The available capacity for relocation of the VM, either to another compatible virtual PM as a stack-entinty, or an actual physical PM. The destination container has to provide the required HWREQ and STACKREQ of the VM, which has to be compatible with the HWCAP and STACKCAP ot the target.
- **SERIALNUMBER|SERNO**
An arbitrary serial number for the VM stored in the configuration file. This number should be unambigiuous.
- **SERVERACCESS|SPORT|S**
Server access port for execution of a TCP/IP connect. This is the raw port to be used for server specific admin tools, which is different from user's client access. This port is currently rarely supported, namely not utilized due to security reasons, e.g. in case of XEN.

The main intention of ctys is to avoid propriatery interfaces as much as possible, and support "bare support tools" only. This interface could only be propriatery. So being honest, 'do not really like that!
- **SPEC_GEN|SPEC**
Generates table output format where each attribute is on a seperate line. This format is particularly forseen to check values with tight reference to the documentation. Refer to common format of generic tables for additional information.
- **SSHPORT**
A list of provided SSH ports on this interface. Currently supported for OpenSSH only.
- **SORT[:<sort-args>]**
Enables the post-sort filter.

<sort-args>=[ALL|A|EACH] [% (UNIQUE|U)] [% (REVERSE|R)] [%<sort-key>]
 - **REVERSE|R** Activates a final filter for call of "sort -r".
 - **UNIQUE|U** Activates a pre-final filter for call of "sort -u".
 - **<sort-key>** Defines a sort key as "-k" option for "sort -k <sort-key>".
- **STACKCAP|SCAP**
The capacity offered by the hypervisor to nested VMs.
- **STACKREQ|SREQ**
The capacity required by the hypervisor as a nested VM itself.
- **STYPE|ST**
Output of the session type, either of category VM, PM, or a HOST by
- **TAB_GEN|TAB**
Generates table output format. The default format could be and is configured as a custom macro. Refer to common format of generic tables for additional information.

- **TCP|T**
The ip address of the VM in stored format.
ATTENTION: In case of multiple interfaces and/or addresses for each address of a so called "multi-homed" machine a seperate entry is generated, thus it is listed as a seperate host entry.
- **TITLE**
The title for any selected field within the output.
- **TITLEIDX**
The title with the related indexes as required and enumerated for input into the generic table.
- **TITLEIDX**
The title with the related indexes as required and enumerated for input into the generic table. In addition the ASC-II values of column indexes for common spreadsheet forms are displayed.
- **USERSTRING|USTRG**
A free editablecustomizable string from the user.
- **UID**
User ID created this record.
- **UUID|U**
Output of UUID.
- **VCPU**
The number of pre-assigned VCPUs.
- **VERSION|VERNO|VER**
Version of config.
- **VMSTATE|VSTAT**
The configured state of the VM. Current auppoted values are: ACTIVE, BACKUP.
- **VNCBASE**
Base port for calculations of ports from display and vice versa. The default is 5900.
- **VNCDISPLAY|DISP**
DISPLAY to be used by XClients, which in case of VNC is already calculated by usage of context-specific PortOffset.
- **VNCPORT|CPORT**
Client access port for execution of a TCP/IP connect. This is the raw port to be used for vncviewer or proprietary clients with their own MuxDemux-dispatcher.
All configured VNC access ports for any VM could be listed as:

```
ctys-vhost -o cport,1 -M all '59\[0-9\]\[0-9\]
```

Where a standard baseport of 5900 is assumed.

VRAM

The amount of pre-assigned VRAM.

XML_GEN|XML

Generates output format in XML format for ease of postprocessing. Refer to common format of generic tables for additional information.

-p <db-directory-path-list>

Comma seperated path list to directories containing the name-resolution DBs, same for each <db-directory-path> as for ctys-vdbgen.

ctys-vhost could handle multiple mapping-DBs for virtual concatenation. The advantage of this is the ability of substructuring VMs and PMs into access-groups by ctys-vdbgen and using them in combinations as required during runtime. This offers particularly advantages when performing ctys-vhost for loadbalancing by usage of cost-option "-C".

-R <remote-hosts>

List of remote hosts for execution. Either a list of valid hostnames, ipaddresses, or EMail-Format hostnames.

-r

Activates the common usage of dynamic runtime data. Without this option only some distinct functions like load-distribution utilize selective calls of runtime-data-evaluation for further restricting their intermediate results. This is e.g. obviously the count of actual executed instances on a PM for the case of cost evaluation on a potential distribution target.

When runtime data evaluation is activated in general, the "-R" option applies to any result as a further constraint.

The usage of runtime data evaluation cost performance of course. This could become dominant, when huge clusters are evaluated, thus should be considered whether really required, and applied to reasonable sets only. But anyhow, when some bigger sets are required by definition, caching of data with different strategies could be applied.

-s

Set when ctys-vhost is used as an internal subcall for another master-tool. In this case some automatic triggered tasks such as the time-driven rebuild of caches are suppressed. Instead a hint for required re-sync is printed as warning.

Urgent tasks will be worked out, even if they might take some minutes. This is the case when no cache is present, of the caches differ in their age. All tools using this as an internal system call should set this flag.

-S <BasicDataManagement>

The "-W" option represents some basic management interfaces for the additional entity class GROUPS and the entity characteristics CONTAINMENT. Where the containment is applied to the whole set of stored entities.

These interfaces allow some smart listing and display of current supported data, the handling of data as deletion and creation is handled by the ctys-vhost command as appropriate.

```
<BasicManagementSupport>=
(
  LISTALL
  |LIST
  |LISTDB
  |MEMBERSDB
  |LISTGROUP[:<groups-list>]
  |MEMBERSGROUP([2345678])|([678]u)[:<groups-list>]
)

<groups-list>:=<group>[,<groups-list>]
```

The following keywords may be applied.

- **LISTALL**

Displays a list of all current available data sources.

- **LIST**

Displays a list of all current data sources, the same as

```
LIST = LISTDB + LISTGROUP
```

- **LISTDB**

Displays a list of current file-databases.

- **MEMBERSDB**

Displays a list of all current staticCacheDB members in ctys-stacked-address notation.

- **LISTGROUP[:<group-list>]**

List all current groups from the CTYS_GROUPS_PATH. The output format is as follows:

```
" <size> <#lrec>/<#incs> <#srec> <group>"
```

- <size> Size n kBytes.
- <#lrecs> The overall number of target entities without resolution of nesting, so just the current file is evaluated.
- <#incs> The overall number of include-statements contained within current file.
- <#srecs> The overall number of target entities with resolution of all nested includes.
- <group> The name of current group, which is the filename too. When "-X" option is set (LEFT of this option), than the basename is shown only, else the full filepathname.

- **MEMBERSGROUP[:<group-list>]**

Lists members of scanned groups. When no <group-list> is provided, the variable CTYS_GROUPS_PATH is decomposed and similar to the PATH variable, any resulting directory is scanned for all existing group files. The members of found groups are displayed.

The nested containment hierarchy by "include" is expanded before output. In case of provided <group-list> the listed groups are displayed only. The format of <group-list> is:

```
<group-list> =: <group-name>[%<group-list>]
```

Two types of storage are shown:

- Raw group files, which may contain target entities, include-statements and comment lines.
- Cache group files, which contain the whole resolved set of containment tree as flat target entity recorded from the statCacheDB.

- **MEMBERSGROUP[2-8][[:<group-list>]**

The values MEMBERSGROUP[2-5] display the same as MEMBERSGROUP, but with a slightly different output format, whereas MEMBERSGROUP[6-8] shows the contained accounts or hosts.

- MEMBERSGROUP2 - List of resulting targets with specific context options, for screen display.
- MEMBERSGROUP3 - Similar to MEMBERSGROUP2, but additionally formatted with line-breaks.
- MEMBERSGROUP4 - List of resulting targets with specific context options, prepared for inclusion by cut-and-paste.
- MEMBERSGROUP5 - List of resulting calls prepared for usage by cut-and-paste.
- MEMBERSGROUP6 - List of accounts in EMail format for screen display only.
- MEMBERSGROUP6u - Unique list of MEMBERSGROUP6.
- MEMBERSGROUP7 - List of accounts in EMail format for machine processing.
- MEMBERSGROUP7u - Unique list of MEMBERSGROUP7.
- MEMBERSGROUP8 - List of hosts for machine processing.
- MEMBERSGROUP8u - Unique list of MEMBERSGROUP8.

Types to be recognized when calculating target. For additional information refer to "-T" option of ctys.

Shows version.

-x <runtime states>

Restricts a set of multiple results with additional constrains for output.

Only the possible targets which are actually operational are listed. This includes the actual running VM with it's hosting PM, and in addition all other operational machines, where the current VM is available too. This case is the most common for NFS based exec-pools, where a single VM could be accessed remotely by a number of PMs. This particularly offers the advantage of copyless-migration of online and offline VMs.

Very handy, and in companion with others probably one of the most important internal top-level-calls for GuestOS-Command-Execution.

```
<runtime states>
  =[MARK|(REVERSE|R|-),]PING|SSH[,PM|VM]
```

- **MARK**

A match for any of the following keywords is simply made with a prefix as running by "R;", instead of just showing the resulting set.

The remaining are formatted with the prefix "-;" for alignment.

- **REVERSE|R|-**

This reverses the resulting set, thus the "not matching" states only will be shown.

- **PING**

A RUNNING session is simply "ping-ed".

Resulting statement of operational mode may result on applied security policies, but not exclusively on the state of the running OS's IP-stack.

- **SSH**

A RUNNING session is additionally checked for SSH-connect by a dummy-interconnect. This might be really senseful, once SSO is established.

"ssh" is the only and one state, which is a viable confirmation for the ability of establishing ctys connections.

- **PM**

Checks only PM for accesibility, which is the default behaviour. PM accesibility is defined as the accesibility of the running OS on PM.

- **VM**

Checks VM for accesibility, this is particularly related to the SSH key. VM accesibility is defined as the accesibility of the running OS on VM.

-X

See ctys, terse for machine output.

REMARK: Due to order dependency of options evaluation, set this as first/leftmost option.

5 ARGUMENTS

Use "-I 2" option for some performance analysis of order dependency for multiple-selection queries.

```
<awk-regexp>[ <awk-regexp>[ <...>]]
```

A list of simple awk regular expression, for matching based on \$0. This is called here "flat-matching", though no structural information like in case of attribute-value assertion, is recognized for the pattern match.

The given lists are matched each on the resulting set of complete records from the previous pattern-matching. The last filter applied will be accompanied by reduction of fields of final matching records as selected by "-o" option.

The main advance of this approach is the simplicity of data structures and the utilization of common tools and data structures. Some performance gain is another advantage.

The drawback is, that in some cases the regexr has to be chosen thoroughly. The first is to suppress shell-expansion for the characters to be passed to the internal awk-expression. These might be required in rare cases only, but offer some advantage.

Some Examples:

- `.` (a single dot)
All items within the database, which is default when missing at all.
- `inst`
All items which contain any string "inst"
- `^inst`
All items, which start with "inst", where the first field in a record is the hostname.
- `*inst`
All items, which end with "inst".
- `xen|qemu`
All items containing 'xen' or 'qemu'.

The next point to be aware of is the order of precedence for the logical operators AND and OR. This is evaluated from **left-to-right**, the **precedence is inverted**. This means, that lower precedences are grouped together by virtual braces. This academic sounding remark offers tremendous positive practical impacts. First the theory - as a result the following rules are identical.

`x AND a OR b OR c`

is equal to

`a OR b OR c AND x`

These could be written as

`x AND a OR x AND b OR x AND c`

Current version does not support braces, but if - the previous could be written as

`x AND (a OR b OR c)`

The next consequence is the repetition of the rules when a higher valued operator AND follows a lower valued group.

`x AND a OR b OR c AND y`

This again could be written as

`x AND (a OR b OR c) AND y`

Or better as

`(x AND (a OR b OR c)) AND y`

This is consequently equal to

`y AND (x AND (a OR b OR c))`

Which could be written as

`y AND x AND (a OR b OR c)`

The main advance of the previous theory is the tremendous simplification and size-reduction of vast powerful and fast scanners. Thus within just a few lines a complete scanner with a recursive logic could be implemented as present within the 'ctys-vhost' command.

The call

```
ctys-vhost -o sort:1 \
  tst00 and not f:2:PM and tst103 or tst00 or tst00
```

or

```
ctys-vhost -o sort:1 \
  not f:2:PM and tst103 or tst00 or tst00 and tst00
```

selects the following set

label	styp	accel	dist	distrel	os	osrel	PM	if	TCP
tst003	VMW		SuSE	9.3	Linux	2.6	delphi.soho	0	172.20.2.133
tst003	VMW		SuSE	9.3	Linux	2.6	delphi.soho	0	172.20.2.133
tst005	VBOX	HVM			Windo		lab02.soho		
tst005	VBOX	HVM			Windo		lab02.soho		
tst005	VBOX						lab02.soho		
tst005	VBOX						lab02.soho		

from a database with 837 entries.

The actual queries could be visualized for analysis purposes. Therefore the intermediate metadata and the resulting matched records are displayed by the option '-I 2'.

The first call produces the output

```
time ctys-vhost -I 2 -o sort:1 \
  tst00 and not f:2:PM and tst103 or tst00 or tst00
```

START R-Methods

CHECK =/homen/acue/.ctys/db/default

START R/W-Methods

RM RTCACHE =/tmp/ctys.acue/ctys-vhost.20101106184019.cdb

QUERY:cacheDB repetitive:"tst00"

```
.....!!!!X!!!!!!!.x.....x.....x.....x.....x50
.....x.....x.....x.....x.....x.....x100
.....x.....x.....x.....x.....x.....x150
.....x.....x.....x.....!!!!X!!!!!!!.x200
.....x.....x.....x.....x.....x.....x250
.....x.....x.....x.....x.....x.....x300
.....x.....x.....!!!!!!!X.....x.....x350
.....x.....x.....x.....x.....x.....x400
.....x.....x.....x.....x.....x.....x450
!.....x.....x.....x.....x.....x.....x500
!!!!!!!X!!!!!!!X!.....x.....x.....x550
.....x.....x.....x.....x.....x.....x600
.....x.....x.....x.....x.....x.....x650
.....!!x.....x.....x.....x.....x.....x700
.....x.....x.....x!.....x.....x.....x750
.....x.....x.....x!!!!!!!.x.....x800
.....x.....x.....x.....
```

match=68 of total=835

QUERY:cacheDB repetitive:"f:2:PM"

```
.....x.....x.....x.....!..x.....x50
.....!X.....
```

```

match=3 of total=68
QUERY:cacheDB repetitive:"tst103"
...
match=0 of total=3
QUERY:cacheDB repetitive:"tst00"
!!!
match=3 of total=3
QUERY:cacheDB repetitive:""
!!!
match=3 of total=3
QUERY:cacheDB repetitive:"tst00"
!!!
match=3 of total=3
QUERY:cacheDB repetitive:""
!!!!!!
match=6 of total=6
!!!!!!
match=6 of total=6
label |styp|accel|dist|distrel|os |osrel|PM |if|TCP
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
tst003|VMW | |SuSE|9.3 |Linux|2.6 |delphi.soho|0 |172.20.2.133
tst003|VMW | |SuSE|9.3 |Linux|2.6 |delphi.soho|0 |172.20.2.133
tst005|VBOX|HVM | | |Windo| |lab02.soho | |
tst005|VBOX|HVM | | |Windo| |lab02.soho | |
tst005|VBOX| | | | |lab02.soho | |
tst005|VBOX| | | | |lab02.soho | |

real 0m1.176s
user 0m0.696s
sys 0m0.532s

```

which requires less steps due to early AND reduction of the metadata. The second call requires much more steps, this is due to the late AND reduction of the metadata, which causes any OR part to be AND checked on the whole initial set.

```

time ctys-vhost -I 2 -o sort:1 \
not f:2:PM and tst103 or tst00 or tst00 and tst00

```

```

START R-Methods
CHECK =/homen/acue/.ctys/db/default
START R/W-Methods
RM RTCACHE =/tmp/ctys.acue/ctys-vhost.20101106184201.cdb
QUERY:cacheDB repetitive:"f:2:PM"
.....X.....X.....X.....X.....x50
.....X.....X.....X.....X.....X!!!!!!X100
!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X150
!!!!!!!X!!!!!!!X!!!!!!!X!!.....x.....x200
.....X.....X.....X.....X.....x250
.....x.....x.....X!!!!!!!X!!!!!!!X!!!!!!!X300
!!!!!!!X!!!!!!!.x.....X.....x.....x350
.....X.....X.....X.....X.....X!!!!!!!X400
!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X450
!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!.....x500
.....X.....X.....X.....X.....x550
.....X.....X.....X.....X.....x600
.....x.....x.....x.....X!!!!!!!X!!!!!!!X650
!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X!!!!!!!X700
!!!!!!!X!!!!!!!X!!!!!!!.x.....x.....x750
.....!!!!X!!!!!!!X!!!!!!!.x.....x.....x800

```

```

.....X.....X.....X.....
match=349 of total=835
QUERY:cacheDB repetitive:"tst103"
.....X.....X.....X.....X.....x50
.....X.....X.....X.....X.....x100
.....X.....X.....X.....X.....x150
.....X.....X.....X.....X...!.....x200
.....X.....X.....X.....X.....x250
.....X.....X.....X.....X.....x300
.....X.....X.....X.....X.....
match=1 of total=349
QUERY:cacheDB repetitive:"tst00"
.....X.....X.....X.....X.....x50
.....X.....X.....X.....X.....x100
.....X.....X.....X.....X.....x150
.....X.....X.....X.....X...!.....x200
.....X.....X.....X.....X.....x250
.....X...!!.....X.....X.....X.....x300
.....X.....X.....X.....X.....
match=3 of total=349
QUERY:cacheDB repetitive:"tst00"
.....X.....X.....X.....X.....x50
.....X.....X.....X.....X.....x100
.....X.....X.....X.....X.....x150
.....X.....X.....X.....X...!.....x200
.....X.....X.....X.....X.....x250
.....X...!!.....X.....X.....X.....x300
.....X.....X.....X.....X.....
match=3 of total=349
QUERY:cacheDB repetitive:"tst00"
.!!!!!!
match=6 of total=7
QUERY:cacheDB repetitive:""
!!!!!!
match=6 of total=6
!!!!!!
match=6 of total=6
label|styp|accel|dist|distrel|os|osrel|PM|if|TCP
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
tst003|VMW| | |SuSE|9.3|Linux|2.6|delphi.soho|0|172.20.2.133
tst003|VMW| | |SuSE|9.3|Linux|2.6|delphi.soho|0|172.20.2.133
tst005|VBOX|HVM| | | | |Windo| |lab02.soho| |
tst005|VBOX|HVM| | | | |Windo| |lab02.soho| |
tst005|VBOX| | | | | | | |lab02.soho| |
tst005|VBOX| | | | | | | |lab02.soho| |

real 0m1.256s
user 0m0.836s
sys 0m0.480s

```

Another effect which could be seen here is the almost equal processing duration, which shows the startup and display overhead as the dominating factor. The times are in the same range for half of the data as shown for 835 entries.

The seemingly double-display is due to the multiple OR in this inverted-precedence semantics, which is resulting in additional display.

The seemingly double-display is due to the hidden display of the owner, here one is the user 'root.root', the other is 'acue.ldapusers'.

For the following rules and operators the search-and-match strings are case sensitive, the operators are not.

AND

The AND operator is the the same as a simple space-operator(" "), which causes the keyword to be applied as selective filter on the previous intermediate result. The result is matched based on the internal MACHINE format, which might lead to different results than the requested final output format only.

E:<#field0>:<#field1>

Compares two fields given by their canonical numbers. The most important application might be the quer for a specific PM record, where the "netname" has to be matched by "PM", which is the "uname -n" Be aware, that only substrings and equal strings match, for local networks using DNS, the "netname" has an additional point "." at the end, thus order of numbers are significant for a match.

The "\$<field0>" is the canonical number as presented by TITLEIDX .

F:<#field0>:<content-match>

Queries for a specific FIELD with provided number to be compared by awk-function "match(\$<#field0>,<content-match>)". Be aware, that only substrings and equal strings match. The match is **checked literally**, this means that 'pm' is NOT EQUAL 'PM'.

The "\$<field0>" is the canonical number as presented by TITLEIDX .

NOT

The NOT operator replaces the current composite state for the next argument only, operators are skipped. It should be recognized, that the NOT operator replaces only the current state, thus no chained evaluation of previous operators is applied. Anyhow, different operators, which are independant, such as NOT and AND, are superposed.

OR

The OR operator adds to the previous intermediate result a filtered subset of the last "AND-result". This sounds maybe a little strange, but simply said, a number of grouped OR operators just imply a parentheses/brace around all OR-ed elements. The overall operations is simple from-left-to-right.

The reason for omitting group-operators is just simplicity of implementation and grant of a resonable overall performance. When more operators are required, a full set of syntax might be implemented.

6 EXIT-VALUES

0: OK: Result is valid.

1: NOK: Erroneous parameters.

2: NOK: Missing an environment element like files or databases.

7: NOK: Missing cacheDB directory.

8: NOK: Missing stat cache.

9: NOK: Missing groups cache.

10: NOK: Missing "macmap.fdb"

11: NOK: Unambiguity was requested by "-M unique", but query result is ambiguous.

7 SEE ALSO

PMs

ctys-PM(1)

VMs

KVM(1) , *ctys-QEMU(1)* , *ctys-VBOX(1)* , *ctys-VMW(1)* , *ctys-XEN(1)*

HOSTS

ctys-CLI(1) , *ctys-RDP(1)* , *ctys-VNC(1)* , *ctys-X11(1)*

ctys-extractARPlst(1) , *ctys-extractMAClst(1)* , *ctys-genmconf(1)* , *ctys-plugins(1)* , *ctys-vping(1)* , *ctys-vdbggen(1)*

dig(1) , *host(1)* , *ifconfig(8)* , *ether-tool(8)* , *nc(1)* <a.k.a. *netcat*>

8 AUTHOR

Maintenance: <acue_sf1@sourceforge.net>
Homepage: <<http://www.UnifiedSessionsManager.org>>
Sourceforge.net: <<http://sourceforge.net/projects/ctys>>
Berlios.de: <<http://ctys.berlios.de>>
Commercial: <<http://www.i4p.com>>



9 COPYRIGHT

Copyright (C) 2008, 2009, 2010 Ingenieurbuero Arno-Can Uestuenseoz

For BASE package following licenses apply,

for software see GPL3 for license conditions,

for documents see GFDL-1.3 with invariant sections for license conditions,

This document is part of the **DOC package**,

- for documents and contents from DOC package see
'**Creative-Common-Licence-3.0 - Attrib: Non-Commercial, Non-Deriv**'
with optional extensions for license conditions.

For additional information refer to enclosed Releasenotes and License files.

