GENESYS

Configuration Layer Objects Reference Guide

CfgApplication

5/1/2025

Contents

CfgApplication

Description

Applications are the various Genesys software programs that serve a contact center. There are two types of Applications: GUI-based Applications and daemon Applications.

Filter Keys

Filter Name	Type	Description
app_type	int	Type of the application (see CfgAppType). If specified, Configuration Server will return information only about the applications of this type.
tenant_dbid	int	A unique identifier of a tenant. If specified, Configuration Server will return information only about the applications that are associated with this tenant.
is_server	int	An indicator of whether this application can be a server to some other applications. Depending on value specified, Configuration Server will return information only about the application(s) that are either servers or non-servers. See CfgFlag.
server_dbid	int	A unique identifier of an application. If specified, Configuration Server will return information only about the applications that are clients to this application.
primary_server_dbid	int	A unique identifier of an application. If specified, Configuration Server will return information only about the application that is backup to this application.
backup_server_dbid	int	A unique identifier of an application. If specified, Configuration Server will return information only about the

Filter Name	Type	Description
		application that is primary to this application.
app_prototype_dbid	int	A unique identifier of an application prototype. If specified, Configuration Server will return information only about the applications that are based on this prototype.
account_type	int	Type of the object that may be used as an account for a daemon application (see CfgObjectType). Makes sense only if used with the filter account_dbid (see below). If both account_type and account_dbid are specified, Configuration Server will return information only about the applications associated with this account. Such information will only be provided to the clients that have privileges to read access control list of this application.
account_dbid	int	A unique identifier of the object that may be used as an account for a daemon application (see type CfgObjectType). Makes sense only if used with the filter account_type (see below). Makes sense only if used with the filter account_dbid (see below). If both account_type and account_dbid are specified, Configuration Server will return information only about the applications associated with this account. Such information will only be provided to the clients that have privileges to read access control list of this application.
host_dbid	int	A unique identifier of a host. If specified, Configuration Server will return information only about the applications currently assigned to this host.
port	int	A server communication port. If specified, Configuration Server will return information only about the applications currently registered at ports with this number. Consider using this filter with filter host_dbid (see

Filter Name	Type	Description
		above).
state	int	Current state of an application (see CfgObjectState). If specified, Configuration Server will return information only about applications that are currently in this state.
name	string	Name of an application. Shall be specified as a character string. If specified, Configuration Server will return information only about the application with that name.
dbid	int	A unique identifier of an application. If specified, Configuration Server will return information only about this application.
same_host_and_port	int	Configuration Server will return information only about applications currently registered on same host and port. Filter is intended to avoid the configuration collisions.
switch_dbid	int	A unique identifier of the switch. If specified, Configuration Server will return information only about T-Servers/HAProxies associated with this switch (see flexibleProperties above). The filter makes sense for application types T-Server and High Availability Proxy (CFGTServer; CFGHAProxy.)
version	string	A version of the application. Shall be specified as a character string. If specified, Configuration Server will return information only about applications with that version.
no_switch_dbid	int	If specified, Configuration Server will return information only about T-Servers/ HAProxies that are not associated with any switches (see flexibleProperties above). The filter makes sense for application types T-Server and High Availability Proxy (CFGTServer; CFGHAProxy.)
no_client_dbid	int	If specified, Configuration Server will return information only about applications/servers which do not have any clients (there is no

Filter Name	Type	Description
		connection to this applications) configured.
startup_type	int	Startup type of the application (see CfgStartupType). If specified, Configuration Server will return information only about the applications of this startup type.

Attributes

- DBID — An identifier of this object in the Configuration Database. Generated by Configuration Server and is unique within an object type. Identifiers of deleted objects are not used again. Read-only.

- name — A pointer to the name of the application. Mandatory. Must be unique within the Configuration Database.

- type — Type of the application. Mandatory. Set automatically according to the value of type of the application prototype specified in appPrototypeDBID or explicitly during the creation time. Once specified, cannot be changed. See CfgAppType.

- password — A pointer to the application password. Max length 64 symbols. Not used in 5.1.

- version — A pointer to the application version. Mandatory. Set automatically according to the value of version of the application prototype specified in appPrototypeDBID or explicitly.

- appServerDBIDs — A pointer to the list of structures of type CfgConnInfo. CfgConnInfo includes:

  - the identifier of the server this application shall connect to as a client when it is started

  - this specific connection properties.

    Applications can be added to the appServerDBIDs structure only if:

  - they are daemons, and

  - their hostDBID and port have been specified.

> **Tip**
>
> Regarding backward compatibility, for applications using Configuration Libraries release 5.1.xxx and 5.1.5xx, the appServerDBIDs will include the list of ServerDBIDs only.

- tenantDBIDs — A pointer to the list of identifiers of the Tenants that are served by this application. Makes sense only for applications of the daemon type. For applications of CFGTServer and CFGHAProxy type, can contain only one tenant. A tenant can be added to this list only if the account that the application is associated with has at least read-only access to this tenant. When used as an entry in CfgDeltaApplication, it is a pointer to a list of identifiers of the tenants added to the existing list.

- isServer — An indicator of whether this application can be a server to some other applications. Read-only (set automatically according to the value of type above). See CfgFlag.

- serverInfo — A pointer to the structure containing server-specific information. Can be specified if, according to the value specified for the type above, the application is a daemon and must be set to NULL otherwise. Once specified, cannot be set to NULL. See CfgServer.

- options — A pointer to the list of application-specific configuration options (see the comments below). When used as an entry in CfgDeltaApplication, it is a pointer to a list of options added to the existing list.

- state — Current object state. Mandatory. Refer to CfgObjectState.

- userProperties — A pointer to the list of user-defined properties. Parameter userProperties has the following structure: Each key-value pair of the primary list (TKVList *userProperties) uses the key for the name of a user-defined section, and the value for a secondary list, that also has the TKVList structure and specifies the properties defined within that section.

- appPrototypeDBID — A unique identifier of an application prototype this application is based on. Optional. See CfgAppPrototype. The association with application prototype could be specified at moment of creation of application object only.

- flexibleProperties — A pointer to the list of additional properties. See comments section at the end of this article. Only described below options can be added to this property. This field can not be changed as long as this server remains a backup for some other server (see CfgServer structure) and can only be non-empty for the applications of type CFGTServer and CFGHAProxy

- workDirectory — Working directory for the application. Must be specified if, according to the value specified in isServer property, the application is a server and optional otherwise. See comments.

- commandLine — The name of executable to be used to start the application. Must be specified if, according to the value specified in isServer property, the application is a server and optional otherwise. See comments.

- autoRestart — Indicates whether the application should be automatically restarted by Local Control Agent after its crash. Mandatory. Recommended to be set to CFGTrue by default. See comments. See CfgFlag.

- startupTimeout — A period of time within which the application is expected to be completely started. Must be specified if, according to the value specified in isServer property, the application is a server and optional otherwise. Default value is 90 seconds. See comments.

- shutdownTimeout — A period of time within which the application is expected to be completely shut down. Must be specified if, according to the value specified in isServer property, the application is a server and optional otherwise. Default value is 90 seconds. See comments.

- redundancyType — Defines the HA type if this application is considered as server. Mandatory. Default is CFGHTColdStandby. See comments. See CfgHAType.

- isPrimary — A role of application within HA/redundancy group. Must be considered in association with redundancyType property. Default is CFGTrue. Read-only (set in accordance with the current role within the HA/redundancy group). See comments. See CfgFlag.

- startupType — A type of application startup. Indicates whether this application have to be started by Management Layer. See type CfgStartupType. Read-only. The property is accessible via API only and not shown by Configuration Manager. Specified during application prototype definition automatically according following: The value is associated with application type CfgAppType. The value for the applications of CFGDBServer and CFGApplicationCluster type is set to CFGSUTDisabled, for other applications of server type is set to CFGSUTAutomatic. For the applications of non-server type is set to CFGSUTDisabled.

- commandLineArguments — A pointer to the additional arguments to be used to start the application. Optional.

- portInfos — A pointer to the list of structures of type CfgPortInfo containing information about listening ports for this Server application. When used as an entry in CfgDeltaApplication, it is a pointer to a list of

port infos added to the existing list.

- resources — A pointer to the list of the objects associated with this Application (every item of this list is structured as CfgObjectResource). When used as an entry in CfgDeltaApplication, it is a pointer to a list of resources added to the existing list. Only objects of type CfgScript can be associated with Application object through resources.

Comments

Deletion of Application(Server) X will cause the following events set out in the order of arrival:

- Modification of detect event of the alarm conditions which referred on Application X

- Modification of backupServerDBID field of the application that had Application X as a backup server

- Modification of appServerDBIDs field of all applications that had Application X in their connections

- Modification of campaign groups of all campaigns that had statServerDBID or dialerDBID fields set to Application X

- Modification of IVR objects that had IVRServerDBID field set to Application X

- Modification of solution and host objects that had SCSDBID field set to Application X

- Deletion of Application X

An application/server cannot be deleted as long as it associated with at least one client application i.e., the connection between client and server is specified within configuration object or it is included to at least one non-optional solution component, or it is assigned as DBServer to at least one table access.

Parameter options has the following structure: Each key-value pair of the primary list (TKVList *options) uses the key for the name of a configuration section, and the value for a secondary list, that also has the TKVList structure and specifies the configuration options defined within that section. Each key-value pair of the secondary list uses the key for the name of a configuration option, and the value for its current setting. Configuration options can be defined as variables of integer, character, or binary type. Names of sections must be unique within the primary list. Names of options must be unique within the secondary list.

> ### Tip
> Configuration Server is not concerned with logical meanings of application-specific configuration sections, options, or their values.

Applications of the daemon type are allowed to establish one and only one communication session to Configuration Server.

Access privileges of an application of the daemon type are determined by the access privileges of the account it is associated with. By default, a new application of the daemon type is associated with access group System (see comments to CfgAccessGroup in section Access Control Functions and Data Types). Function ConfSetAccount can be used to change the default account.

Access privileges of an application of the GUI type are determined by the access privileges of the currently logged-on person. See comments to CfgPerson.

An application is allowed to establish a communication session with Configuration Server only if the currently logged-on person (for GUI applications) or the account (for daemon applications) has Execute permission with respect to this Application (see type CfgACE).

An application of CFGConfigServer type with DBID = 99 shall be pre-defined (scripted) in the Configuration Database before Configuration Server is started for the first time. The object that represents this application cannot be deleted.

An application of CFGSCE type with DBID = 100 shall be pre-defined (scripted) in the Configuration Database before Configuration Server is started for the first time. The object that represents this application cannot be deleted.

An application can be included into different solutions. Configuration Server does not provide the synchronization property tenantDBID of CfgService and property tenantDBIDs of CfgApplication. The tenantDBIDs list should be updated manually or by wizard every time a solution the application is a part of is assigned to a new tenant (the corresponding tenant's id should be added to the list). Similar (manual or by wizard) update should be made if a solution the application is included into is no longer associated with a tenant (the corresponding tenant's id should be removed from the list).

An application can not be deleted as long as there is at least one solution the application is a part of.

After upgrading from CME 5.1.x to 6.1 the following default values should be set for the application:

- workDirectory = '.'
- commandLine = name (the value of name property)
- autoRestart = CFGFalse
- startupTimeout = 90
- redundancyType=CFGHTColdStandby
- isPrimary=CFGTrue
- startupType= <should be set according to description of startupType above>

An application cannot be deleted if it has a type CFGITCUtility (53).

The name of the application can not be changed if there is, at least, one active client exist registered under this name.

XML Representation

> **Tip**
> This XML was created using the Configuration Server 7.5 schema.

```
<CfgApplication>
        <DBID value="100" />
        <name value="myName" />
        <password value="the_password" />
        <type value="19" />
        <version value="7.5" />
        <isServer value="1" />
        <state value="1" />
        <appPrototypeDBID value="100" />
        <autoRestart value="0" />
        <startupTimeout value="0" />
        <shutdownTimeout value="0" />
        <redundancyType value="0" />
        <isPrimary value="0" />
        <startupType value="3" />
</CfgApplication>
```

See Also

CfgDeltaApplication

CfgHost