



Inventario y CMDB



om:

<https://pandorafms.com/manual/!776/>

permanent link:

https://pandorafms.com/manual/!776/es/documentation/10_pandora_itsm/11_pandora_itsm_inventory

2024/06/10 14:34



Inventario y CMDB

[Volver al índice de documentación de Pandora FMS](#)

Inventario

La funcionalidad de Inventario proporciona todas las utilidades necesarias para gestionar todo el conjunto de objetos de los que dispone la compañía, conocer su ubicación, su fabricante, sus contratos de soporte, las personas y teléfonos de contacto, referencias y números de serie, etcétera. Los objetos de inventario pueden ser asociados a *tickets*, permitiendo una gestión eficiente de los recursos materiales e intangibles de la compañía.

Dentro del inventario existen diferentes tipos de objetos, con campos propios. Es posible crear nuevos tipos y modificar, añadir o borrar campos de cualquier tipo de objeto.



Creación de un tipo de objeto de inventario

Los tipos de objeto de inventario sirven para personalizar los diferentes campos que el objeto tendrá. Puede gestionar los tipos de objetos pulsando en la opción CMDB → Object types (Tipos de objetos).



Para editar un tipo de objeto haga clic en su nombre (columna Name):



Un tipo de objeto de inventario tiene asociado un nombre, un icono, el valor mínimo que debe tener en estocaje (*stock*) y una descripción. Además podrá definir si este tipo aparecerá o no como raíz en la vista de árbol del inventario.

Campos personalizados

Cada tipo de objeto tiene una serie de campos personalizados. Por ejemplo, el tipo de objeto de inventario llamado Software tiene el campo asociado Versión (*Version*):



Para agregar más campos tales como Ubicación (*Location*) y Fabricante (*Software developer*), pulse en el botón Create y rellene los campos como se explica en “[Propiedades de campos personalizados](#)” y presione de nuevo el botón Create.



Al existir varios campos personalizados podrá cambiar el orden relativo entre ellos con los botones de flechas en la columna de acciones (Actions). También podrá borrar cada uno en dicha columna por medio del icono de papelera.

Propiedades de campos personalizados



- Label: Etiqueta del objeto.
- Types: Tipo de información a rellenar luego, los campos pueden ser:
 - *Numeric* (numérico).
 - *Checkbox* (casilla de verificación).
 - *Text* (texto).
 - *Combo* (lista de opciones): Al ser seleccionado se habilitará un cuadro de texto llamado Combo values donde debe introducir los diferentes valores separados por comas (,).
 - *External* ([campo externo](#)).
 - *Date* (fecha).
- Unique: Si esta opción es activada, solamente podrá haber un objetos de inventario con el mismo valor para este campo (similar a un identificador o ID).
- Global field: Si esta opción es activada, se permite agregar este campo personalizado a todos los tipos de objetos. *Úsese con precaución.*
- Inherit: Si esta opción es activada, se heredarán los valores de propietario, usuarios y empresas del *objeto padre*.
- Show in list: Si esta opción es activada, se muestra en el listado general por defecto.
- Prevent updates: Si esta opción es activada, se el campo *no se actualizará* al recibir datos de inventario remoto.

Campos de tipo externo

Los campos de tipo externo mostrarán información de tablas adicionales (añadidas manualmente por un administrador) en la base de datos de Integria IMS.

Para configurar y poder utilizar un campo de tipo externo se necesita crear un nuevo objeto de inventario con el Tipo Externo seleccionado y aparecerán todos los campos necesarios para su

configuración:



- External table name: Nombre exacto de la tabla externa, en la base de datos, de la tabla donde se encuentra este campo externo. *Obligatorio*.
- Field to be displayed: Campo que se visualizará, nombre de la columna que contiene los valores que se mostrarán finalmente en el objeto de inventario. *Obligatorio*.

Para la creación de un tipo externo siendo padre de otro objeto externo:

- Primary key: Campo identificador o “clave primaria”, normalmente corresponderá a la columna ID. Solo es necesario si se trata de una tabla/campo padre.

Para la creación de un de tipo externo siendo hijo de otro objeto externo:

- Foreign key: Campo de la tabla hijo que relaciona ambas tablas, corresponde a la Foreign Key en la tabla hijo. Solo es necesario si se trata de una tabla/campo hijo.
- Parent table name: Nombre de la tabla padre. Solo es necesario si se trata de una tabla/campo hijo.

Con esta configuración ya se podrá emplear un campo de tipo externo. A la hora de crear o modificar un objeto de inventario de un tipo que contenga campos externos, aparecerá una ventana modal mostrando la información de la tabla externa:



Además, los campos de tipo externo se pueden relacionar entre sí, creando dependencias. Para ello las dos tablas externas a utilizar en nuestros dos campos externos deben también estar relacionadas por alguna de sus columnas, que habitualmente será un ID. Habitualmente la tabla padre tendrá una columna ID que será su `primary key` y la tabla hijo tendrá una columna tipo `ID_relación` que corresponderá a su `foreign key`, columna que relacionará esta tabla hijo con su tabla padre. Esto hará que dependiendo del valor que seleccione en el primer campo, el segundo campo muestre valores diferentes.

Caso práctico

Se tiene una tabla añadida manualmente, `t_ubication`, con dos columnas: `id` y `ubication`, el primero es el identificador único, el segundo el valor útil a mostrar.



Para configurar un campo de tipo externo que haga uso de la información contenida en esta tabla se configura del siguiente modo:



Así se utiliza en primer lugar el nombre de la tabla `t_ubication`, y en segundo lugar la columna `ubication` a utilizar en el campo de tipo externo llamado `Ubication`.



Una vez hecho esto, al crear un objeto de inventario del tipo que contenga este campo externo se deben seguir los siguientes pasos:

- Creación de objeto de inventario en el campo externo:



- Clic en el icono de lupa y se elige un valor de la tabla externa:



- Tras seleccionar una de las ubicaciones mostradas en el cuadro de diálogo emergente (`Madrid`) estará seleccionada en el objeto.



- Ahora se podrá crear un *segundo* campo de tipo externo que estará relacionado con el campo `Ubication` utilizado. Los pasos son similares, pero completando los campos adicionales `Primary key` en el campo padre (`Ubication`), y `Foreign key` y `Parent table name` en el campo hijo (`Location`).



- Debe crea una tabla llamada `t_location`, con tres columnas: `id`, `location` e `id_ubication`.
- Ahora debe completar los campos correspondientes para relacionar esta tabla con la anterior. En el campo padre añade la clave primaria:



- Y debe crear el campo hijo del siguiente modo:



Se ha configurado los campos `Primary key` en la tabla padre con su columna `id`, y `Foreign key` y `Parent table name` con la columna de clave extranjera `id_ubication` de la tabla hijo, y el nombre de su tabla padre.

En caso de que haya seguido los pasos tal y como en el ejemplo, habrá creado una relación de dependencia, y en función del valor que elija en el primer campo personalizado, en el segundo tendrá unas u otras posibilidades.

En este ejemplo hemos elegido `Madrid` en el campo `Ubication` y aparecerá `Atocha` como única opción elegible al ser la única que existe con el `id_ubication` igual a 1:



El mantenimiento de estas tablas externas se realiza de forma manual operando con ella por medio de sentencias SQL, o bien, utilizando el editor de tablas externas que

proporciona Integria IMS.



Edición de tablas externas

En el listado de tipos de objeto de inventario, aquellos que tengan campos de externos contarán con un icono para edición de tablas.



Sobre estas tablas se pueden realizar operaciones de edición y borrado de datos existentes, así como añadir datos nuevos.



Creación de un objeto de inventario

Desde el menú lateral izquierdo se accede desde CMDB → Create new object.



Los campos más importantes en este formulario son:

- Owner (Propietario).
- Associated companies (Compañías asociadas).
- Associated users (Usuarios asociados).
- Public (Público).

Estos campos definen quién puede ver el objeto: De esta forma el objeto estará accesible por el propietario, los usuarios asociados directamente al objeto o los usuarios pertenecientes a una compañía asociada. Además si se marca la *flag* Público todos los usuarios podrán ver dicho objeto.

Un objeto de inventario puede tener una “jerarquía” de objetos (un objeto puede ser hijo de otro). Esto se define al escoger un *padre*. Además de este parentesco *padre e hijo*, se le pueden establecer relaciones entre objetos.

El inventario de Integria IMS posee un sistema de control de *stocks*. Para gestionar el *stock*, todos los objetos de inventario tienen un campo “estado” que permite llevar un sistema de *stock* (también se pueden registrar la fecha de recepción y baja). Los estados posibles son:

- New (Nuevo).
- In use (En uso).
- Unused (No usado).
- Issued (Dado de baja).
- Además de todos aquellos definidos por el usuario en el menú de configuración del inventario de **Pandora FMS**.

Ejemplo:



La pestaña de relaciones muestra los demás objetos relacionados con este. Se pueden asociar nuevos objetos, solamente hay se escoge de la lista y se vincula como se muestra a continuación. De esta forma, se pueden relacionar otros objetos además del objeto *padre*.



La pestaña de contactos muestra las personas asociadas a ese objeto de inventario. Haciendo clic en cualquiera de ellos se podrá acceder a su información de manera directa.




La solapa de *tickets* asociados al objeto de inventario indica qué *tickets* ha sufrido este objeto, y en cuál estado se encuentran. Se puede hacer clic en cualquiera de ellas para acceder a su información de manera directa.



La pestaña *tracking* (Seguimiento) muestra los cambios que se producen en el objeto de inventario tales como el cambio de propietario, si se asocia un tipo de objeto, un objeto *padre*, etcétera.



TRACKINGInventory object #31: 

Description	User	Date
Updated	Default Admin	2021-11-08 11:55:36
Owner changed -> Old: 'Default Admin' - New: 'Jaime Blanco'	Default Admin	2021-11-08 11:55:36
Inventory object type added -> Software	Default Admin	2021-11-08 11:55:36
Parent changed: -> Old: 'sample_linux_host' - New: 'sample_linux_host'	Default Admin	2021-11-08 11:55:36
Parent added -> sample_linux_host	Default Admin	2021-11-08 11:55:36
Users updated	Default Admin	2021-11-08 11:55:36
Companies updated	Default Admin	2021-11-08 11:55:36
Updated	Default Admin	2021-01-21 19:45:42
Inventory object type added -> Software	Default Admin	2021-01-21 19:45:42

Búsquedas de inventario

Los campos por los que se pueden realizar búsquedas son:

- Search (Buscar): Campo de texto libre que hace referencia al nombre o descripción del inventario si está activo el *token*. Esto es así si no se seleccionan campos personalizados. En este caso, la búsqueda acota también al valor de estos campos.
- Search for description (Buscar por descripción): Añadir al campo Search la descripción del objeto.
- Status: Estado del objeto.
- Block size por pagination: Selecciona el tamaño del bloque de paginación. Por defecto esto se configura en las **opciones generales** y se limita entre 2 y 1000.
- Associated company: Compañía asociada
- Contract: Contrato.
- Parent object: Objeto padre.
- Owner: Propietario.
- Associated user: Usuario asociado.
- Tipo de objeto. Es posible seleccionar más de un tipo de objeto en la búsqueda mediante el selector
- Last updated: última actualización.
- Object type: Tipo de objeto.



Además puede exportar los resultados de las búsquedas en formato CSV. El resultado de la búsqueda de inventario puede visualizarse en dos modos: árbol o listado.

En la vista de árbol se muestran los objetos de inventario agrupados por tipo. También se visualiza

el *stock*, mostrándose:

- Unidades totales.
- Nuevas.
- Sin usar.
- *Stock* mínimo. Si el número de unidades totales es inferior al *stock* mínimo, se marcará en rojo.



En la vista en modo listado se muestra información sobre el inventario, así como sus campos personalizados que hayan sido marcados para mostrar.



Importar datos de inventario desde CSV

Para ello se *carga* el fichero CSV con una cabecera y una línea por cada objeto de inventario. La cabecera debe estar en la primera fila de nuestro CSV y debe contener al menos la columna *name* ya que el nombre del objeto de inventario debe existir.

- *name* (Nombre): Obligatorio. Se tendrá en cuenta la opción del *setup* que permite tener **nombres de objetos de inventario duplicados**.
- *owner* (Propietario): Esta columna recoge los identificadores de los usuarios que sean propietarios del objeto, los cuales deben de existir en la base de datos. Si no se introduce esta columna el valor introducido por defecto estará vacío.
- *id_parent* (Objeto padre): En esta columna irá el identificador del objeto de inventario padre, los cuales deben de existir en la base de datos. Por defecto es cero (0).
- *id_manufacturer* (Empresa): En esta columna irá el identificador de las empresas a la que pertenece cada objeto de inventario, con lo cual estas empresas deben de existir en la base de datos. Por defecto es cero (0).
- *id_contract* (Contrato): En esta columna irá el identificador del contrato al que pertenece cada objeto de inventario. Por defecto es cero (0).
- *status* (Estado): En esta columna se pueden recoger los cuatro tipos de estado que define por defecto Integria IMS (*new*, *inuse*, *unused*, *issued*) o un estado definido por el usuario.
- *receipt_date* (Fecha de recepción): En esta columna irá la fecha de recepción y deberá tener un formato de fecha válido.
- *issue_date* (Fecha de uso): En esta columna irá la fecha de uso y deberá tener un formato de fecha válido.
- *description*
- *public*: Solo admitirá valor 0 ó 1 dependiendo si quiere que sea público o no. Por defecto es uno (1).
- *id_company* (Compañías asociadas) las compañías asociadas pueden ser una o varias, por lo tanto introduciremos los ids de la siguiente manera: [*id1* .*id2* .*id3* .*id4*]. Es importante el uso de corchetes y dentro de ellos el uso puntos para separar los elementos.
- *associated_user* (Usuarios asociados): Los usuarios asociados pueden ser uno o varios, por lo tanto introduciremos los ids de la siguiente manera: [*id1* . *id2* .*id3* . *id4*]. Es importante el uso de corchetes y dentro de ellos el uso puntos para separar los elementos.

- `id_object_type` (Tipo de objeto): En esta columna irá el identificador de los tipos de objetos los cuales deben de existir en la base de datos. Por defecto es cero (0).
- El resto de columnas que se añadan corresponderán a los campos personalizados de un tipo de objeto, existen cuatro tipo de campos diferentes los cuales deberán respetar las limitaciones de dichos campos:
 - *Campos de tipo texto*: No tienen ninguna limitación.
 - *Campos de tipo numérico*: Sólo permiten números, por lo tanto la columna de este campo deberá contener solamente números.
 - *Campos de tipo combo*: Permiten una lista de palabras, por lo tanto los caracteres introducidos en esta columna sólo podrán ser los que se recogen en dicha lista.
 - *Campos de tipo external*: Estos campos permiten asociar tablas externas para introducir los valores se validará que exista dicha tabla y sus referencias.

Todos los campos personalizados también deberán cumplir si son únicos o no. Esto depende de si ha seleccionado la creación de los campos personalizados, con lo cual esto provocará no poder introducir valores que ya existan en estos campos personalizados.

Ejemplo de fichero CSV editado en una hoja de cálculo:



Inventario remoto basado en agentes de Pandora FMS

Esta funcionalidad Enterprise permite extraer información de inventario personalizada de cualquier dispositivo y sistema operativo.

El inventario remoto se basa en el uso de los **agentes software de Pandora FMS que son instalados** en las máquinas de las que se desea extraer la información. Los agentes recolectan información de inventario mediante *scripts* y envían los datos al servidor donde se está ejecutando Integria IMS. Una vez recibida la información IIMS la procesará y creará los objetos de inventario pertinentes en base a los datos recibidos.

A diferencia de la Sincronización de inventario con Pandora FMS, para usar el Inventario remoto basado en agentes de Pandora FMS no es necesario instalar la aplicación Pandora FMS, sólo es necesario instalar los agentes en las máquinas que se desee.

Funcionamiento interno del inventario remoto

Los agentes de Pandora FMS envían un fichero XML con la información de inventario al servidor que soporta Integria IMS. IIMS procesa estos archivos mediante el *script* de mantenimiento: el procesado de archivos generalmente se realiza cada 5 minutos. Estos XML deben recibirse en el

directorio /attachment/inventory ubicado dentro del directorio de IIMS.



Con la información de inventario enviada en el XML se crearán varios objetos de inventario. De tal forma que si un agente envía información sobre 12 aplicaciones instaladas en el sistema, se crearán 13 objetos de inventario (uno para el agente y 12 para las diferentes aplicaciones instaladas).

Cuando se recibe información sobre un dispositivo o elemento de inventario, pueden suceder dos cosas:

1. Que el elemento exista: Entonces se actualizará la información de inventario.
2. Que el elemento no exista: En este caso, se creará un nuevo objeto de inventario sólo si el tipo de objeto asociado está definido en Integria.

Es importante definir los tipos de objeto que usarán antes de comenzar a realizar el inventario, a continuación se explica cómo funciona la adquisición de datos en detalle y podrá ver cómo se asocian los tipos a los diferentes elementos.

Los agentes de Pandora FMS están asociados al tipo llamado Pandora agents, por lo que para que el inventario funcione es necesario que esté definido. Los demás objetos de inventario se asocian a sus tipos cómo se explica a continuación.

Ejemplo XML:

```
<?xml version='1.0' encoding='UTF-8'?>
<agent_data description='' group='' os_name='linux' os_version='"CentOS release
6.4
(Final)'" interval='300' version='7.0NG' timestamp='2021/10/22 09:27:38'
agent_name='sample_linux_host' timezone_offset='0' address='192.168.70.163'>
<inventory>
<inventory_module>
<name><![CDATA[Users]]></name>
```

```

<datalist>
  <data><![CDATA[root]]></data>
</datalist>
</inventory_module>
<inventory_module>
  <name><![CDATA[Process]]></name>
  <datalist>
    <data><![CDATA[COMMAND]]></data>
    <data><![CDATA[/sbin/init]]></data>
    <data><![CDATA[[kthreadd|]]></data>
    <data><![CDATA[[migration/0|]]></data>
    <data><![CDATA[[ksoftirqd/0|]]></data>
    <data><![CDATA[[scsi_eh_6|]]></data>
    <data><![CDATA[sh -c ps -eo command | tr -d " " ]]></data>
    <data><![CDATA[ps -eo command]]></data>
    <data><![CDATA[tr -d ]]></data>
  </datalist>
</inventory_module>
<inventory_module>
  <name><![CDATA[Software]]></name>
  <datalist>
    <data><![CDATA[acl;2.2.49;Access control list utilities]]></data>
    <data><![CDATA[acpid;1.0.10;ACPI Event Daemon]]></data>
    <data><![CDATA[alsa-lib;1.0.22;The Advanced Linux Sound Architecture
(ALSA)
library]]></data>
    <data><![CDATA[alsa-plugins-pulseaudio;1.0.21;Alsa to PulseAudio
backend]]>
    </data>
    <data><![CDATA[zenity;2.28.0;Display dialog boxes from shell
scripts]]></data>
    <data><![CDATA[zip;3.0;A file compression and packaging utility compatible
with PKZIP]]></data>
    <data><![CDATA[zlib;1.2.3;The zlib compression and decompression
library]]>
    </data>
  </datalist>
</inventory_module>
<inventory_module>
  <name><![CDATA[File system]]></name>
  <datalist>
    <data><![CDATA[tmpfs;0;244M;/dev/shm]]></data>
    <data><![CDATA[/dev/sda1;34M;426M;/boot]]></data>
  </datalist>
</inventory_module>
</inventory>
</agent_data>

```

IIMS creará un ítem para el agente que reportó el XML y a continuación procesará los diferentes módulos de inventario por separado, por ejemplo:

```

<inventory_module>
  <name><![CDATA[Software]]></name>
  <datalist>
    <data><![CDATA[acl;2.2.49;Access control list utilities]]></data>
    <data><![CDATA[acpid;1.0.10;ACPI Event Daemon]]></data>
    <data><![CDATA[alsa-lib;1.0.22;The Advanced Linux Sound Architecture
(ALSA)
library]]></data>
    <data><![CDATA[alsa-plugins-pulseaudio;1.0.21;Alsa to PulseAudio
backend]]>
</data>
    <data><![CDATA[zenity;2.28.0;Display dialog boxes from shell
scripts]]></data>
    <data><![CDATA[zip;3.0;A file compression and packaging utility compatible
with PKZIP]]></data>
    <data><![CDATA[zlib;1.2.3;The zlib compression and decompression
library]]>
</data>
  </datalist>
</inventory_module>

```

Se escogerá el nombre del módulo de inventario definido en el XML, en este caso Software, para identificar el tipo de objeto asociado a estos elementos.

Después, se crearán ítems de inventario con dicho tipo para cada uno de los elementos reportados en el XML:

```

<datalist>
  <data><![CDATA[acl;2.2.49;Access control list utilities]]></data>
  <data><![CDATA[acpid;1.0.10;ACPI Event Daemon]]></data>
  <data><![CDATA[alsa-lib;1.0.22;The Advanced Linux Sound Architecture (ALSA)
library]]></data>
  <data><![CDATA[alsa-plugins-pulseaudio;1.0.21;Alsa to PulseAudio backend]]>
</data>
  <data><![CDATA[zenity;2.28.0;Display dialog boxes from shell scripts]]></data>
  <data><![CDATA[zip;3.0;A file compression and packaging utility compatible
with PKZIP]]></data>
  <data><![CDATA[zlib;1.2.3;The zlib compression and decompression library]]>
</data>
</datalist>

```

Los elementos de inventario separan la información por el carácter ; . La información quedaría desglosada como se puede ver a continuación:

```

<data><![CDATA[acl;2.2.49;Access control list utilities]]></data>

```

```

acl      2.2.49      Access control list utilities

```

El primer componente define el nombre para el nuevo objeto de inventario. El resto de los componentes estarán asociados a los campos de tipo de objeto en el mismo orden en el que

aparecen en la definición de objetos del inventario. En este caso el segundo elemento se asocia a Version y el tercero a Description.



Además, todos los objetos correspondientes a los elementos tendrán como padre el agente que los reportó a través del XML. El resultado final para el objeto de inventario creado para ese elemento del XML se verá como en esta imagen:



En el inventario, junto con todos los objetos que se han creado basándose en los distintos módulos de inventario del XML, un ítem aparecerá para el agente, en este caso llamado `localhost.localdomain`



Por defecto, Integria IMS tiene los módulos de inventario principales predefinidos para agentes MS Windows®, tales como Software, Parches, HD, Vídeo, CPU, Servicios y RAM:



Configuración del inventario remoto

La configuración del inventario automático se encuentra en el menú Setup en la pestaña **Pandora FMS Inventory**, sección Remote Inventory.

Los campos configurables son:

- Default owner: Dueño por defecto del nuevo objeto de inventario.
- Associated company: Compañías por defecto para el nuevo objeto de inventario.
- Associated user: Usuarios por defecto para el nuevo objeto de inventario.

El campo Default owner es obligatorio, si no se configura no se crearán los objetos de inventario. Los demás campos sirven para propósitos de configuración de ACLs de acceso y visualización de los objetos.

Además de la configuración de los campos mencionados es necesario habilitar el *script* de mantenimiento. Puede ver cómo instalar este *script* en la sección “Instalación del *script* de mantenimiento”.

Si no habilita el *script* de mantenimiento no se procesarán los archivos enviados por los agentes de Pandora FMS.

El envío de archivos se hace mediante una conexión TCP al puerto 41121. Por ello para el correcto funcionamiento de esta funcionalidad puede que tenga que revisar las reglas de *routing* o *firewalls* de red.

Configuración de los agentes de Pandora FMS

Para extraer la información de los dispositivos deberá [instalar los agentes de Pandora FMS](#). Puede ver la compatibilidad del agente de Pandora FMS con sus sistemas en la sección "Requisitos agente software del manual de Pandora FMS".

Los parámetros adicionales a configurar en el agente son los siguientes:

- `server_ip`: Dirección IP del servidor que ejecuta Integria IMS.
- `server_path`: Directorio de recepción de archivos del servidor.
- `module_plugin`: Directiva para ejecutar *scripts* para extraer la información de inventario.

Por defecto los agentes de Pandora FMS disponen de un módulo *plugin* para extraer información de inventario:

```
module_plugin inventory 1 cpu ram video nic hd cdrom software
init_services filesystem users process ip route
```

Personalización del inventario

Si desea personalizar la información recogida puede modificar este *script* o crear uno propio.

En la sección "[Creación de módulos de inventario locales](#)" del manual de Pandora FMS se describe la estructura XML de los módulos de inventario y ejemplos de cómo crear módulos de inventario para plataformas GNU/Linux® y MS Windows®.

[Volver al índice de documentación de Pandora FMS](#)