



News > AX 2012 > AX 2012 Release Notes > AX 2012 5.0.0.3 and Lasernet 10 Release Notes

## AX 2012 5.0.0.3 and Lasetnet 10 Release Notes

2023-10-20 - Alex Clemons - Comments (0) - AX 2012 Release Notes



### AX 2012 v.5.0.0.3

**Release date:** 16th September 2022

#### Release

[Click to download the LAC2012 v.5.0.0.3 installation package.](#)

#### Note

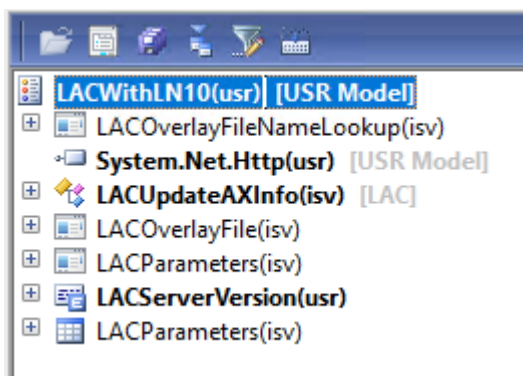
5.0.0.3 is an XPO file with changes to existing LAC objects and new objects. Installing LAC 5.0.0.3 requires that you have already installed LAC version 5.0.0.1 or 5.0.0.2.

#### Features

This page outlines the features of the Lasetnet Connector for AX2012.

#### Import of Model

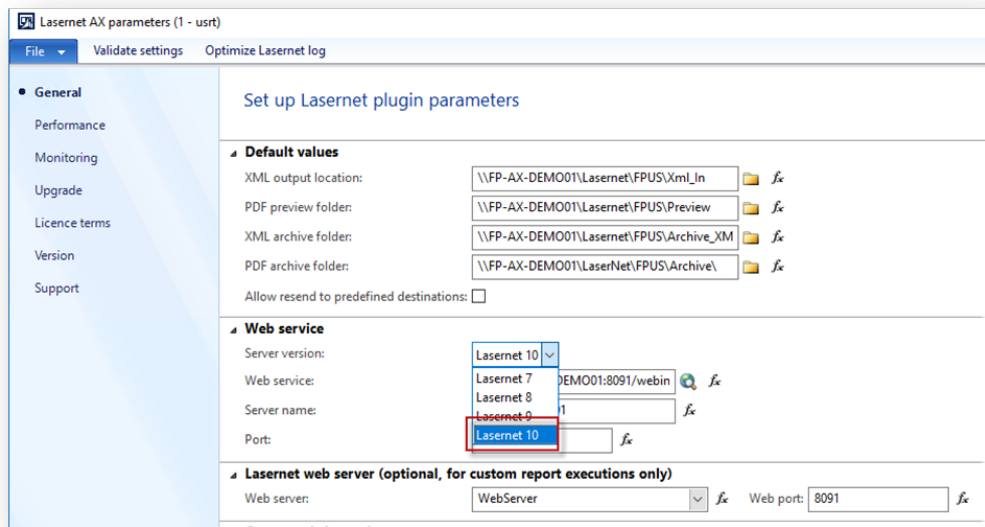
The Project/XPO for LAC2012 contains the following objects for import:



#### Adjustment of Lasetnet Connector for AX2012

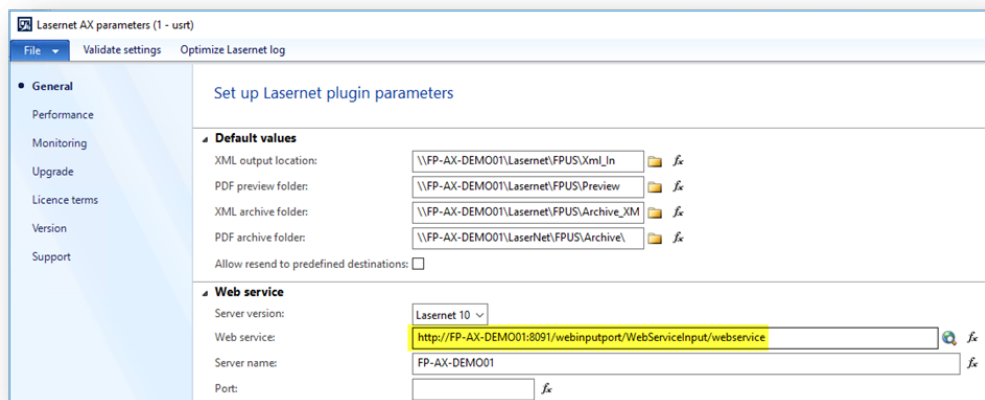
When setting up the plugin parameters, ensure that Lasetnet 10 is

selected. Lasernet 7, 8, and 9 should also function in the same way, but version 10 should be selected in case Lasernet 10 is used.

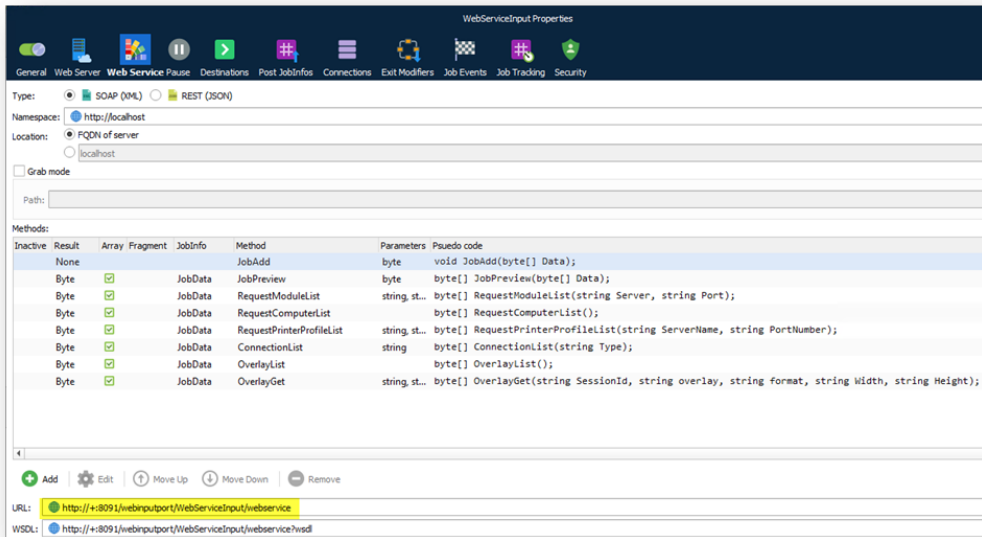


## Configuration of Web Service and Web Server

The web server can be configured in the box for Web service.



This will correspond to the following web service URL as it appears in Lasernet:



## Web Service

Navigate to the following page to access the server:

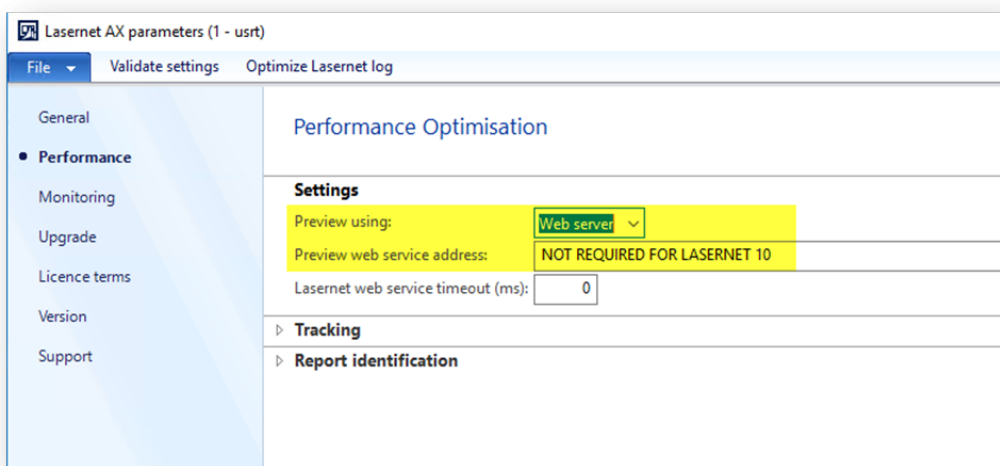
<http://<ServerName>:<Port>/webinputport/WebServiceInput/webservice/>

Example:

<http://FP-AX-DEMO01:8091/webinputport/WebServiceInput/webservice/>

## Web Server

Select the Web server by clicking the **Preview using** drop-down menu.



## Adjustments to the Lاسernet Configuration

### Scripts

A range of new scripts are available with this package for retrieving Overlays, Custom port(s), Printer(s), and Printer profile(s).

After installation, the scripts will appear in the Lasernet interface by clicking the **Scripts** button in the left-hand panel:

#### Gets Overlay List for Dynamics 365 FO

```
function LN10GetOverlayList()
{
    var overlays = configuration.getOverlays();
    var result = '<LaserNet>';
                //'<OverlayListResponse>';
    for (idx = 0, count = overlays.length; idx < count; idx++)
    {
        result = result + '<ln:string>'+overlays[idx]
+ '</ln:string>';
    }
    result = result + //'</OverlayListResponse>'+
                '</LaserNet>';
    job.setJobData(result);
}
```

#### (LaserNet 10) Gets Overlay Thumbnail for Dynamics 365 FO

```
function LN10GetOverlay(overlayFileName, overlayWidth, overlayHeight)
{
    var result = '<LaserNet>';
    result = result +
'<ln:OverlayGetResult>'+Base64.encode(configuration.getOverlayThumbnail(overlayFileName, overlayWidth, overlayHeight)) +
'</ln:OverlayGetResult>';
    result = result + '</LaserNet>';
    Logger.logEvent(0,configuration.getOverlayThumbnail(overlayFileName,
overlayWidth, overlayHeight));
    job.setJobData(result);
}
```

#### (LaserNet 10) Gets Module List for Dynamics 365 FO

```
function LN10GetModuleList()
```

```

{
    var result = '<LaserNet>';
    var modules = configuration.getModules();
    for (idx = 0, count = modules.length; idx < count; idx++)
    {
        result = result + '<ln:Module>';
        result = result + '<ln:Name>'+modules[idx].name+'</ln:Name>';
        if (modules[idx].type == 'Printer Service') // This is for
printer services
        {
            result = result + '<ln:Type>'+Printer
Output'+</ln:Type>';
        }
        else
        {
            result = result +
'<ln:Type>'+modules[idx].type+'</ln:Type>';
        }
        result = result +
'<ln:Description>'+modules[idx].description+'</ln:Description>';
        result = result +
'<ln:Inactive>'+!modules[idx].active+'</ln:Inactive>';
        if (modules[idx].printer)
        {
            result = result +
'<ln:Printer>'+modules[idx].printer+'</ln:Printer>';
        }
        if (modules[idx].printserver)
        {
            result = result +
'<ln:PrintServer>'+modules[idx].printserver+'</ln:PrintServer>';
        }
        result = result + '</ln:Module>';
    }
    result = result + '</LaserNet>';
    Logger.logEvent(0, result);
    job.setJobData(result);
}

```

**(Lasernet 10) Gets Computer List for Dynamics 365 FO**

```

function LN10GetComputerList()
{
    var result = '';
    var servers = configuration.getServers();
    for (idx = 0, count = servers.length; idx < count; idx++)
    {
        Logger.logEvent(Debug, 'Server=' + servers[idx].name + '
(Hostname=' + servers[idx].hostname + ', Port='+ servers[idx].port
+')');
        result = result + '';
        result = result + ' + servers[idx].name + '';
        result = result + ' + servers[idx].hostname + '';
        result = result + ' + servers[idx].port + '';
        result = result + '';
    }
    result = result + '';
    job.setJobData(result);
}

```

### **(Lasernet 10) Gets Connection List for Dynamics 365 FO**

```

function LN10GetConnectionList()
{
    var result = '<LaserNet>';
    var connections = configuration.getConnections();
    for (idx = 0, count = connections.length; idx < count; idx++)
    {
        result = result + '<ln:ConnectionInfo>';
        result = result + '<ln:Name>' + connections[idx].name +
'</ln:Name>';
        result = result + '<ln:Description>' +
connections[idx].description + '</ln:Description>';
        result = result + '<ln:GUID>' + connections[idx].guid +
'</ln:GUID>';
        result = result + '<ln:Type>' + connections[idx].type +
'</ln:Type>';
        result = result + '</ln:ConnectionInfo>';
    }
    result = result + '</LaserNet>';
    job.setJobData(result);
}

```

```
}
```

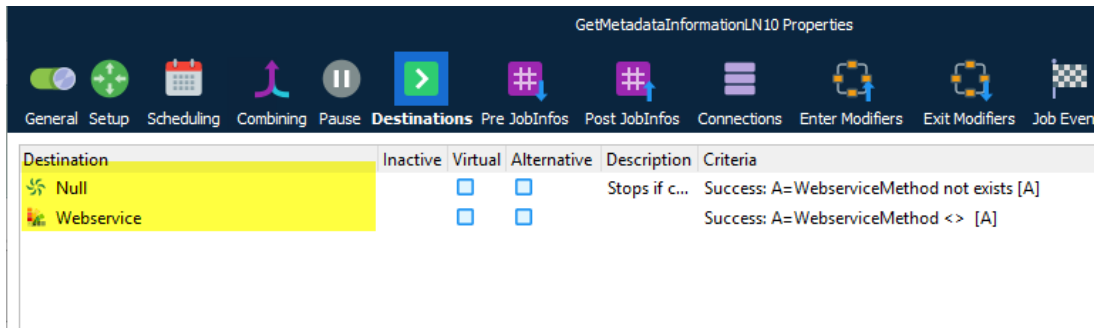
## (Lasernet 10) Gets Printer Profile List for Dynamics 365 FO

```
function LN10GetPrinterProfileList()
{
    var result = '<LaserNet>';
    var printerprofiles = configuration.getPrinterProfiles();
    for (idx = 0, count = printerprofiles.length; idx < count; idx++)
    {
        result = result + '<ln:Printer>';
        result = result + '<ln:Name>' + printerprofiles[idx].name +
        '</ln:Name>';
        result = result + '<ln:PrinterName>' +
printerprofiles[idx].printername + '</ln:PrinterName>';
        for (idy = 0, county = printerprofiles[idx].profiles.length;
idy < county; idy++)
        {
            result = result + '<ln:Profile>';
            result = result + '<ln:Name>' +
printerprofiles[idx].profiles[idy].name + '</ln:Name>';
            result = result + '</ln:Profile>';
            //    Logger.logEvent(Debug, 'PrinterProfile=' +
printerprofiles[idx].profiles[idy].name);
        }
        result = result + '</ln:Printer>';
    }
    result = result + '</LaserNet>';
    job.setJobData(result);
}
```

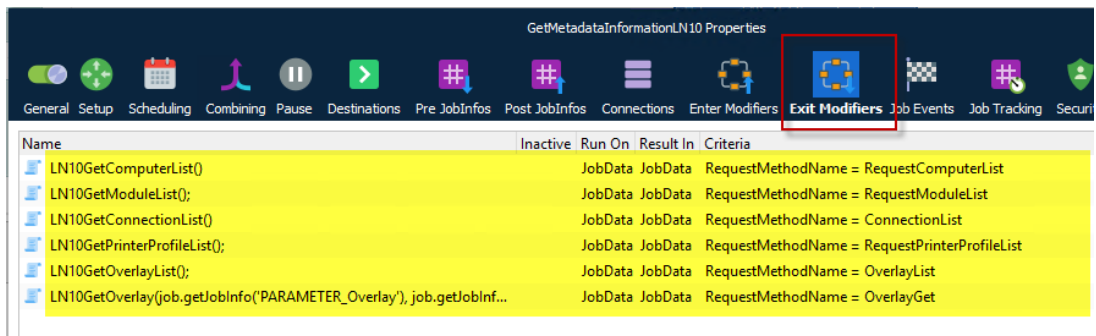
### **New Module: GetMetaDataInformationLN10**

The module **GetMetadatalInformationLN10** which retrieves Lasernet metadata now comes with several corresponding Destinations and Exit Modifiers, as follows:

### **Destinations**



## Exit Modifiers

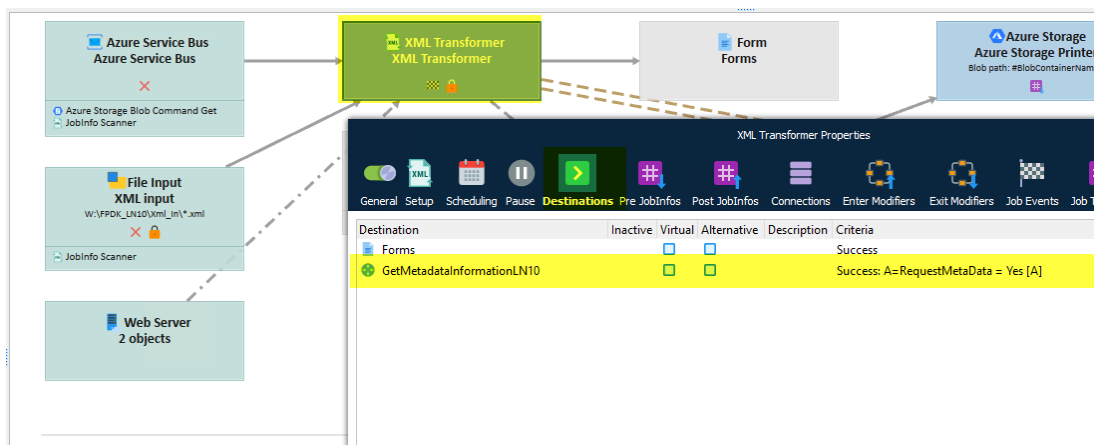


## XML Transformer

The destination **GetMetadataInformationLN10** is added to the XML Transformer.

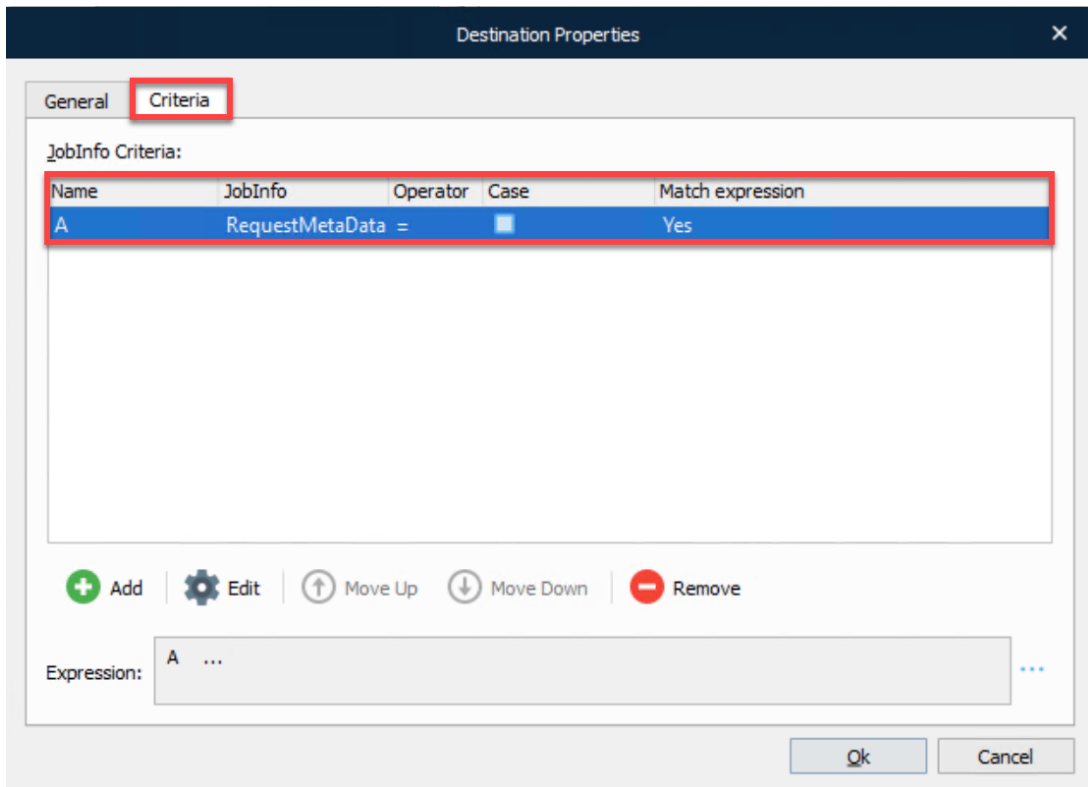
Destination: **GetMetadataInformationLN10**

Criteria: **RequestMetaData**



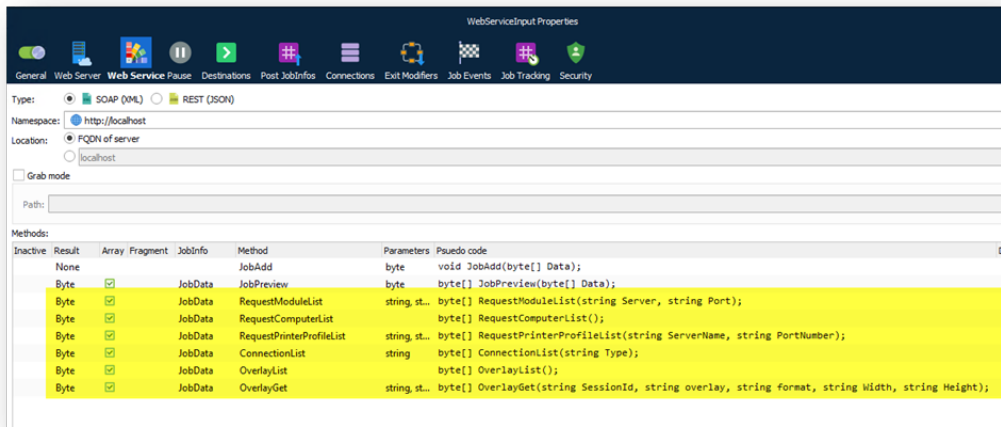
Criteria for the **GetMetadataInformationLN10**:





### Adjustment to WebServiceInput

The following new methods has been added to the module **WebServiceInput**:



### RequestModuleList

Method Properties

Active:

Name: RequestModuleList

Description: Optional

Result: Byte + Add... ⚙ Edit...

Name	Data Type	Reference	Array	Fragment	JobInfo
Server	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_Server
Port	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_Port

+ Add
⚙ Edit
⬆ Move Up
⬇ Move Down
- Remove

Pseudo code: `byte [] RequestModuleList(string Server, string Port);`

## RequestPrinterProfileList

Method Properties

Active:

Name: RequestPrinterProfileList

Description: Optional

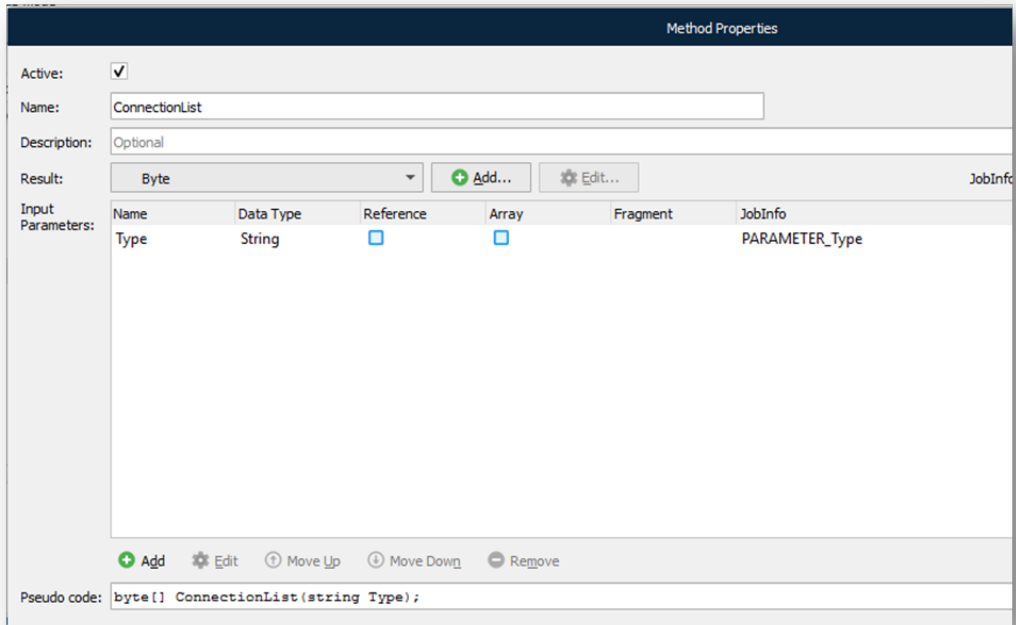
Result: Byte + Add... ⚙ Edit...

Name	Data Type	Reference	Array	Fragment	JobInfo
ServerName	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_ServerName
PortNumber	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_PortNumber

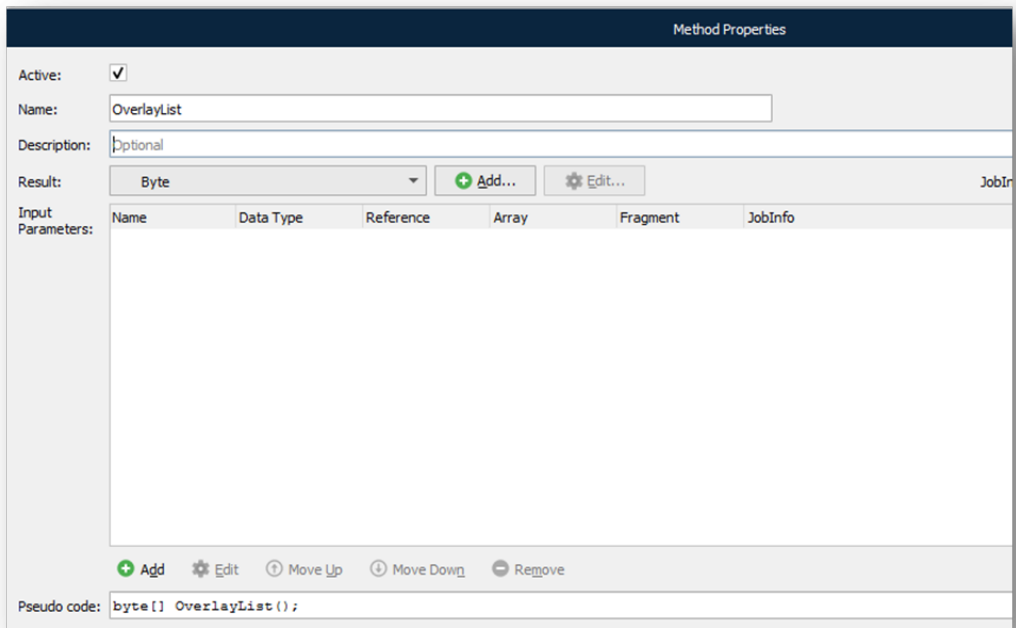
+ Add
⚙ Edit
⬆ Move Up
⬇ Move Down
- Remove

Pseudo code: `byte [] RequestPrinterProfileList(string ServerName, string PortNumber);`

## ConnectionList



## OverlayList



## OverlayGet

Method Properties

Active:

Name: OverlayGet

Description: Optional

Result: Byte + Add... ⚙ Edit... JobInfo: JobData

Name	Data Type	Reference	Array	Fragment	JobInfo	Description
SessionId	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_SessionId	
overlay	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_Overlay	
format	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_Format	
Width	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_Width	
Height	String	<input type="checkbox"/>	<input type="checkbox"/>		PARAMETER_Height	

+ Add ⚙ Edit ↶ Move Up ↷ Move Down − Remove

Pseudo code: `byte[] OverlayGet(string SessionId, string overlay, string format, string Width, string Height);`

## Further Information

If you encounter difficulty while using Lasernet Connector for AX2012, please contact us via the [Create Ticket](#) button in the support portal.

## Related Content

- [AX 2012 Version 5.0.0.3 Released!](#)
- [AX 2012 with LAsernet 10.xpo](#)