Installing LANSA on Linux

- About this Guide
- Introduction
- LANSA for Linux Installer Overview
- Install LANSA for Linux
- Customize the LANSA Installation
- Install LANSA Integrator on Linux
- Appendix A. Help Text

Edition Date October 23, 2014 V13 SP2 © LANSA

About this Guide

- This guide provides instructions for installing LANSA software on the Linux platform. During the installation you may install:
 - LANSA for Linux
 - LANSA Integrator
 - the LANSA Apache Module.
- The contents are written for technical support staff and LANSA developers and assume that readers have a solid understanding of both the Linux operating system and LANSA.
- How to install or configure a Linux operating system or any other prerequisite software is beyond the scope of this guide. It is assumed that all the prerequisite software is loaded and configured.
- It is assumed that an experienced System Administrator (root user) of the Linux system is available to carry out system administration tasks and advise the reader on operating system issues.
- It is also assumed that an ORACLE Database Administrator (DBA) is available to create and configure databases, create user ids and advise the reader on ORACLE issues.
- We recommend the use of the Korn shell (ksh) or Bourne shell (sh). All example Linux commands in this guide use the Korn shell.

Additional Information

For more details about LANSA for Linux, refer to these guides:

Deploying LANSA Applications on Linux Guide

LANSA Communications Setup Guide

LANSA Technical Reference Guide

For the latest product information, refer to the LANSA product web site at http://www.lansa.com/support/v13news

1. Introduction

On Linux, LANSA provides a server execution environment for LANSA applications. It does not provide a development environment or a GUI execution environment for LANSA applications.

- 1.1 LANSA for Linux Main Functions
- 1.2 Definitions and Terminology
- 1.3 Root Privileges Required

1.1 LANSA for Linux Main Functions

LANSA for Linux has the following main functions:

- Receive objects delivered from Visual LANSA importing the repository, creating tables, building the shared libraries (the equivalent to Windows DLLs) for OAMs and other LANSA objects required on the server. For more information, refer to Deliver the Server Portion of an Application to Linux in the Deploying LANSA Applications on Linux Guide.
- The LANSA listener (lcolist): waits for incoming requests from client applications, and starts jobs (lcotp) to service those requests. For more information on the LANSA listener, please refer to the *LANSA Communications Setup Guide*.
- Command line execution (x_run): can be used directly from the Linux command shell to execute batch jobs, check license installation, and so on.
 On Linux, all x_run jobs are automatically in batch mode as there is no user interface. For more information on x_run, please refer to The X_RUN Command in the *Technical Reference Guide*.
- LANSA for the Web Run-time: supports running LANSA applications developed using the LANSA Web Functions and Web Application Modules technology. For more information, please refer the LANSA for the Web Guide.
- LANSA for the Web Apache Module: the Apache Web Server add-on for serving web pages generated by LANSA systems on the same Linux system and/or any other additional LANSA systems, including those on Windows or IBM i. It supports both LANSA Web Functions and Web Application Modules. For more information, please refer the LANSA for the Web Guide.
- LANSA Integrator for Linux: enables integration of Application-to-Application (A2A) and Business-to-Business (B2B) transactions through XML and Java Services and optionally using LANSA applications on the same Linux system and/or any other additional LANSA systems, including those on Windows or IBM i. For more information, please refer the LANSA Integrator Guide.

Please refer to the LANSA guides listed in Additional Information for further details on LANSA for Linux.

1.2 Definitions and Terminology

The LANSA Group

The LANSA Group is a valid Linux group id that is used to set group ownership for the whole LANSA installation directory except a few special files and subdirectories. In order to support allowing multiple Linux user ids to use the same LANSA installation at the same time, LANSA for Linux is designed to work relying on user group permissions. Except for a few special operations (see below about Root privileges usage), any program execution, including for example running x_run from the command line and server type processes spawned off from the Listener, from the LANSA installation directory must be running using a valid Linux user id belonging to the LANSA Group. For the normal course of running LANSA applications from the installation directory in a production environment, various files and directories will be created and accessed in the installation directory. When new files or directories are created, they will be usually created such that they are readable and writable by the LANSA Group. Existing files or directories, including those created manually, to be accessed the LANSA applications in use must be readable and/or optionally writable (depending on the LANSA applications in question) by the LANSA Group.

For multiple LANSA installations on the same Linux system, there is no need to use the same LANSA Group across all the installations. Using different Linux group ids as the LANSA Group for the multiple installations can actually reduce potential interference between the multiple installations.

Unless you have very special requirements, you should avoid using any Linux system group id, for example "bin" and especially "root", as the LANSA Group. It is highly recommended to create a specific group id for use as the LANSA Group.

The LANSA Group must be provided to the LANSA for Linux installer upon installation.

The LANSA Owner

The LANSA Owner is a valid Linux user id that is used to set user ownership for the whole LANSA installation directory except a few special files and sub-directories. The LANSA Owner must be a member of the LANSA Group and preferably the LANSA Group is being the primary group of the LANSA Owner. While the LANSA Owner can be used to run LANSA applications from the installation, it is not a requirement to run LANSA applications using the

LANSA Owner only. Any Linux user id belongs to the LANSA Group (and preferably as the primary group) would be able to run LANSA applications from the installation. It is perfectly all right that the LANSA Owner is not used to run anything at all. However, some crucial configuration files are modifiable only by the LANSA Owner. You will need to login as the LANSA Owner to make changes to those files.

For multiple LANSA installations on the same Linux system, there is no need to use the same LANSA Owner across all the installations. Using different Linux user ids as the LANSA Owner for the multiple installations can actually reduce potential interference between the multiple installations.

Unless you have very special requirements, you should avoid using any Linux system user id, for example "bin" and especially "root", as the LANSA Owner. It is highly recommended to create a specific user id for use as the LANSA Owner.

The LANSA Owner must be provided to the LANSA for Linux installer upon installation.

Apache Daemon User and Group

The Apache Daemon User is a valid Linux user id for running the Apache HTTP daemon.

The Apache Daemon Group is a valid Linux group id for running the Apache HTTP daemon.

For example in a default Red Hat Enterprise Linux installation, the defaults for both are usually named "apache".

If you install LANSA for the Web (including the Apache Module and standard images and scripts) and/or LANSA Integrator for Linux, a few special files and sub-directories in the installation directory will be explicitly installed to be owned by the Apache Daemon User and Group. The remaining files and sub-directories for these two areas of installation will still be owned by the LANSA Group and Owner but they will be accessed by the Apache HTTP daemon and so the permissions will be checked against the Apache Daemon User and Group instead of the LANSA Group and User. If you make changes to those files and sub-directories and/or adding files and sub-directories, you may need to make sure the files and sub-directories affected are readable and/or optionally writable by the Apache Daemon User and Group depending on your application.

For LANSA for the Web Apache Module, the Apache Daemon User and Group are not normally used or required to run any LANSA applications from the same or any other LANSA installations, including those not on the same system. In

other words, the execution of the Apache Module is very much independent of the execution of your LANSA applications and security-wise this is a good thing.

For LANSA Integrator for Linux and if you use JSMDirect, LANSA applications running through the LANSA Integrator (effectively the x_run command for the LANSA installation which the LANSA Integrator uses) will be running under the Apache Daemon User and Group and so, as described above about the LANSA Group, the Apache Daemon User must be a member of the LANSA Group of the LANSA installation which the LANSA Integrator uses. You can either manually add the Apache Daemon User to the LANSA Group for the LANSA installation which you use or there is an option for the LANSA for Linux installer to do that for you upon installation. If you have multiple LANSA installations that JSMDirect would use, the Apache Daemon User may need to made member of more that one LANSA Groups.

For standard Apache Web Server setup, LANSA for Linux installer will attempt to figure out the Apache Daemon User and Group names from standard Apache configuration files. If that fails or you have a non-standard Apache Web Server setup, you will need to provide the Apache Daemon User and Group to the LANSA for Linux installer upon installation.

LANSA Table Owners

LANSA table owners (also known as collections) are special ORACLE database user ids (reminded not to confuse with Linux user id) that own LANSA tables. The LANSA table owner LX_DTA exists to own the LANSA repository. Application-specific LANSA table owners exist to own a collection of LANSA tables. They are directly related to the data library associated with a table. For example, the partition data library for the DEM partition shipped with LANSA is DC@DEMOLIB. The associated collection is called XDEMOLIB.

When accessing a table from the database, LANSA automatically prefixes the collection name to the table. For example, SELECT ... FROM TABLE LX_DTA.LX_MSG.

For further details on how collections are named, refer to *Automatic Renaming for SQL/ODBC* in the *LANSA Application Design Guide*.

LANSAXROOT

In the following description, LANSAXROOT is usually used to refer to the directory on the Linux system under which the LANSA installation resides. This is also referred to as the LANSA installation root directory or simply installation root. During normal operations using LANSA for Linux, for either an

interactive shell or any LANSA system process, an environment variable of the same name is usually required to be set to the installation root.

1.3 Root Privileges Required

LANSA for Linux requires root privileges for the following operations only:

- Running the LANSA for Linux installer.
- Starting and stopping the LANSA Connect Listener.
- Starting and stopping the Apache Web Server. (Optional)

For any other LANSA for Linux operations, root privileges are generally not required. Most of the time, you should simply use a login, which is a member of the LANSA Group of the LANSA installation you are using. The login is not even necessarily being the LANSA Owner unless you are going to make changes to certain configuration files.

2. LANSA for Linux Installer Overview

LANSA for Linux Installer can be used to install various combinations of software packages available in the full LANSA for Linux install package.

The installer is called vlinstall.py and is a Python interpreter script. You need the Python interpreter of version 2.4 or later in the version 2 series to run the installer. (Note that Python interpreter of version 3.0 or later is not supported.) For modern Linux systems, for example, Red Hat Enterprise Linux release 5 or later, the Python interpreter of the required version is usually installed by default. You can run the command "python -V" in a command prompt to check if you have the Python interpreter and its version if it is there. If you do not have the Python interpreter on your Linux system, you need to refer to the documentation of your Linux system for installation instructions for the Python interpreter.

The full LANSA for Linux install package includes the following software packages:

- 1. **LANSA Communications**: including the LANSA Listener (lcolist) and other supporting server programs and client libraries (compulsory for all types of installation)
- 2. **LANSA for Linux**: including x_run, the LANSA runtime libraries including support for LANSA for the Web and other supporting programs and libraries
- 3. **LANSA for the Web Apache Module for Linux**: add-on to the Apache Web Server for serving generated web pages from LANSA for the Web runtime
- 4. **LANSA Integrator for Linux** (there are options for the installer to install only the required components of LANSA Integrator)

The installer provides the following installation types:

- **Full install** Installs all the software packages described in 1-4 above.
- LANSA for Linux only (default)
 Installs only LANSA for Linux and LANSA Communications for running various types of LANSA applications including, for example, command line type applications using x_run and applications developed using LANSA for the Web technology that are to be activated by separate LANSA for the Web plug-in installations.
- LANSA for the Web Apache Module for Linux only

Installs only LANSA for the Web Apache Module and LANSA Communications client libraries for serving generated web pages from some separate LANSA systems that can be a separate LANSA for Linux installation on the same or other Linux system or Visual LANSA for Windows or LANSA for IBM i on some separate systems.

To install this installation type, you need to invoke vlinstall.py with the option --web-server-only like this command line:

./vlinstall.py --web-server-only {...other installation options...}
This is equivalent to using the LANSA for the Web Apache Module installer with command line like this:

./wpiinstall.py {...other installation options...}

Refer to the section Install LANSA for the Web Apache Module for Linux below for more information.

LANSA Integrator for Linux only

Installs only LANSA Integrator for Linux only.

To install this installation type, you need to invoke vlinstall.py with the option --integrator-only like this command line:

./vlinstall.py --integrator-only {...other installation options...}
This is equivalent to using the LANSA Integrator for Linux installer with command line like this:

./liinstall.py {...other installation options...}

By default, this installation type installs all available LANSA Integrator for Linux components including JSMServer, JSMProxy, UserAgent and RFI.

To choose only the required components, you need to use the option -- integrator-options in command line like this:

```
./vlinstall.py --integrator-only \
```

--integrator-options=jsmserver,jsmproxy,useragent,rfi $\$

{...other installation options...}

Any combination of the available 4 components can be specified for the option. The option value shown above is effectively the default to install all available components.

Refer to the section Install LANSA Integrator for Linux below for more information.

3. Install LANSA for Linux

If you are installing or upgrading LANSA for Linux, perform the following steps:

- Step 1. Prepare for LANSA installation (requires root privileges)
- Step 2. Prepare database for LANSA (requires DBA privileges)
- Step 3. Execute the install script (requires root privileges)
- Step 4. Optionally Install Licenses (requires LANSA owner or root privileges)
- Step 5. Customize the Install (requires DBA privileges)
- Step 6. Configure LANSA Communications Extensions (requires LANSA owner or root privileges)
- Step 7. Create listener startup and shutdown scripts (requires root privileges)
- Step 8. Install EPCs (requires root privileges)
- Step 9. Start the listener (requires root privileges)

Once you have successfully completed all the installation steps, please go on to Customize the LANSA Installation.

If you have any problems with the install, refer to Troubleshooting in the *Deploying LANSA Applications on Linux Guide*.

Step 1. Prepare for LANSA installation (requires root privileges)

- 1. Optionally, request the appropriate LANSA licenses. Licenses are only required at execution time, so you can request and install licenses after the install. Refer to the LANSA website for information.
- 2. Create a new Linux group for the LANSA installation. This will allow access to the LANSA files and directories to be restricted to users who belong to the group.
- 3. Create a Linux user to 'own' LANSA for Linux. This user id is referred to as the LANSA owner.
- 4. Decide on a directory for the LANSA installation. For example: /opt/lansa. This directory is known as LANSAXROOT. We recommend a bare minimum of 200Mb of free space (to allow for the LANSA installation and the import of the DEM partition).

Step 2. Prepare database for LANSA (requires DBA privileges)

- 1. Install and configure ORACLE Server. Note that this is not necessarily the same machine to be used to run LANSA for Linux.
- 2. Create a database for LANSA (e.g. using dbca) and ensure the Oracle listener is running (using the default port of 1521).
- 3. Note the host name of ORACLE Server, ORACLE SID, and ORACLE userid and password. These information will be provided to the LANSA for Linux installer later.

Step 3. Execute the install script (requires root privileges)

1. Change directory to the install directory and execute the installation script to create the LANSA tables and copy files to their correct locations.

For a default installation, that is, installing LANSA for Linux only, the following command line will check if the provided options are valid:

```
cd <Install image directory>
./vlinstall.py -I <LANSAXROOT> -O <LANSA Owner> -G <LANSA
Group> \
-D <Data source name> -u <Oracle username> -p <Oracle password>
```

-D <Data source name> -u <Oracle username> -p <Oracle password> \
--create-dsn --oracle-server=<Oracle host> --oracle-sid=<Oracle SID>

Oracle Instant Client will be installed immediately if it is found to be missing. The command will report if the options appear to be correct but nothing further will be installed until you run the installation. If an error is reported, adjust the command lines according to the error reported. If there is no error, then start the installation by adding the --Commit option to the command line. The command line becomes:

```
cd <Install image directory>
./vlinstall.py -I <LANSAXROOT> -O <LANSA Owner> -G <LANSA
Group> \
    -D <Data source name> -u <Oracle username> -p <Oracle password> \
    --create-dsn --oracle-server=<Oracle host> --oracle-sid=<Oracle SID> \
    --commit
```

For a full or customised installation, run the following command to see the full list of options for the installer:

```
./vlinstall.py -h
```

For instance, if you install the LANSA for the Web Apache Module or LANSA Integrator for Linux, you will need to provide addition options relating to your current Apache Web Server configuration.

2. Once the install script has been completed, check the output to ensure no problems occurred during the install. Make sure that there are no "RUNSQL ended in error" messages. These messages have a non-zero Return code. For further information, refer to the Troubleshooting notes in the *Deploying*

LANSA Applications on Linux Guide, if necessary.

Step 4. Optionally Install Licenses (requires LANSA owner or root privileges)

Install the LANSA licenses and review their status. Refer to Enter new LANSA License Codes for more information.

Note: You must always stop the listener before entering or removing license codes in x_lickey.dat.

Step 5. Customize the Install (requires DBA privileges)

This step is optional. Please read the instructions below and decide whether you want to configure database-specific settings before the install.

Note: If you change any files listed here, make a copy of the original with the extension .original. Make a copy of your changed file with the extension .changes. This will allow your changes to be reapplied after an upgrade or EPC installation.

Change the default and temporary tablespace (in x_dbmenv.dat)

By default, LANSA tables will go into the tablespace USERS, which by default is quite small. You may wish to increase the size of this tablespace, or create a new tablespace for LANSA. If you choose the latter option, alter the settings used to create the users in \$LANSAXROOT/x_lansa/source/crlansa.ora.

You should copy any changes you make here into \$LANSAXROOT/x_lansa/x_dbmenv.dat. Adjust the following line (under the heading ORACLE - Linux Platforms):

CMD_CREATE_COLLECTION=CREATE USER >C< IDENTIFIED BY >C< DEFAULT TABLESPACE USERS TEMPORARY TABLESPACE TEMP QUOTA UNLIMITED ON USERS

Change the tablespace settings (via X_RUNSQL_XXX environment variables)

LANSA's table creation utility, runsql, creates tables and indexes with ORACLE-specific column types, but without database-specific settings; for example, tablespace, initial extent size, and so on.

If you wish to append database-specific settings to all CREATE TABLE and CREATE INDEX statements issued by LANSA, set the environment variables X_RUNSQL_CREATE_TABLE and X_RUNSQL_CREATE_INDEX in the LANSA owner's login script. Please refer to The RUNSQL Utility in the Deploying LANSA Applications Guide for full details such as, which environment variable affects primary key indexes.

Step 6. Configure LANSA Communications Extensions (requires LANSA owner or root privileges)

The LANSA install creates a default lansa.ini, lroute.dat and listen.dat in the \$LANSAXROOT directory. If you would like to make changes to the defaults, refer to Linux Configuration in the *LANSA Communications Setup Guide*.

Step 7. Create listener startup and shutdown scripts (requires root privileges)

LANSA provides a script, setenv.sh, that sets up the environment variables required for command line LANSA operations using the standard shell 'sh' or 'bash'. If you use another shell, for example, csh, you will need to create your own script based on setenv.sh. This script must be loaded in a shell prompt (only once is required for the whole live-time of the shell prompt) using the shell dot '.' command before running any LANSA command line in the same shell prompt. A shell prompt running with root privileges is required to submit the command lines to start or stop the LANSA listener.

For example, assuming you are using the standard shell 'bash' for Red Hat Enterprise Linux, the command line to load the script looks like:

. /lansa/x_lansa/bin/setenv.sh

Notice the space between the first dot and the following slash is part of the command line and cannot be omitted.

Alternatively, you may set the LANSAXROOT environment variable first:

LANSAXROOT=/lansa; export LANSAXROOT

. \$LANSAXROOT/x_lansa/bin/setenv.sh

If any errors are found during the script's execution, error messages will be issued and the scripts return code will be non-zero.

To start the listener (listening at the default port 4545), issue the following command:

lcolctl start

To stop the listener, issue the following command:

lcolctl stop

For further information, please refer to *Running the Listener Job on the Linux Server* in the *LANSA Communications Setup Guide*.

Note: Before starting the listener, it is always best to issue the lColCtl stop command, just in case the listener is still running, or was accidentally killed.

If you want to start/stop the LANSA Listener upon system starts up/shut down, you will need to create your own system service script based on the information above. For Red Hat Enterprise Linux, for example, the system service script is usually located in the directory /etc/init.d. You may following existing scripts

there to create your own for the LANSA Listener.

Step 8. Install EPCs (requires root privileges)

If there are EPCs (Expedited Program Changes) to be installed, install them now.

- If EPCs are shipped on the CD or DVD containing the LANSA software for Linux, this will be mentioned in the web pages on the CD/DVD.
- At other times, EPCs may be available from your LANSA vendor.

Step 9. Start the listener (requires root privileges)

If you created you own system service script in Step 7. Create listener startup and shutdown scripts (requires root privileges), you can use it to start the LANSA Listener. Reminded you need root privileges to do that.

If you don't want to create your own system service script for any reason, you can always start/stop the LANSA Listener manually according to Step 7.

For details on how to ensure that the listener started correctly, please refer to Running the Listener Job on the Linux Server in the LANSA Communications Setup Guide.

4. Customize the LANSA Installation

Now that you have installed LANSA successfully, review the following topics to decide which of them need to be addressed:

- 4.1 Allow Users Access to LANSA
- 4.2 Modify LANSA Configuration Files and Environment Variables
- 4.3 Start Multiple Listeners
- 4.4 Automate listener startup and shutdown
- 4.5 Back up application data

4.1 Allow Users Access to LANSA

When you first start using LANSA for Linux, you may decide to use the LANSA owner as your Linux login for testing, and your LANSA Table Owner user id(s) to connect to the database.

When you are ready to move to production, you will probably want to create Linux and ORACLE user ids for each user or group. Even if you do not want to have a unique ORACLE and Linux user id for each user, you should create at least one separate Linux and ORACLE user id that only has permissions to execute the application and view and modify data.

- 4.1.1 Add a new ORACLE User Id
- 4.1.2 Add a new Linux User Id
- 4.1.3 Prepare for executing x_run batch jobs

4.1.1 Add a new ORACLE User Id

If you want each user to be distinguishable by their ORACLE user id, you will need to create an ORACLE user id for each user. If not, you should create at least one ORACLE user id to distinguish a user from the owner of the tables. This will allow you to stop users from modifying or dropping required tables. LANSA requires ORACLE user ids and passwords of no more than 10 characters.

When you add a new user to access the LANSA application, grant the role LANSA_USER to the user. This role has been granted SELECT, INSERT, UPDATE, and DELETE privileges on every table and view that LANSA creates. It also has the CREATE SESSION privilege.

4.1.2 Add a new Linux User Id

The user must have permission to read files under the LANSAXROOT directory and subdirectories, and permission to execute the files in the \$LANSAXROOT/x_lansa/bin directory. The easiest way to accomplish these goals is to add the user to the LANSA owner's primary group.

During execution, LANSA may need to create or update files in particular directories. If the user does not have permission to create files in those directories, processing may continue as expected, without the creation of the file (for example, when attempting to produce trace information), or processing may fail. It is important that you determine which directories users need permission to create files in, and confirm that users have appropriate rights.

To help you with your planning, here are some of the default locations for the most common files created during LANSA processing:

- If LANSA Communications Tracing and logging is enabled, lroute.trc and lroute.log are created or updated in the \$LANSAXROOT directory. To change the location, refer to Linux Configuration Files in the LANSA Communications Setup Guide.
- If LANSA internal tracing is enabled (x_run argument ITRO=Y), x_err.log, x_tracennn.txt and temporary files are created in the \$LANSAXROOT/log directory. To change the location, refer to TPTH= in Standard X_RUN Parameters in the *Technical Reference Guide*.
- By default, the .RRN files (which must be in the same location for all users), are created in the partition directory. For example, for the DEM partition, this is \$LANSAXROOT/x_lansa/x_dem. To change the location, refer to RPTH= in Standard X_RUN Parameters in the Technical Reference Guide.
- If PRTR=*PATH, reports are created in the partition directory. For example, for the DEM partition, this is \$LANSAXROOT/x_lansa/x_dem. To change the location, refer to PPTH= in Standard X_RUN Parameters in the *Technical Reference Guide*.

4.1.3 Prepare for executing x_run batch jobs

If the user needs to execute x_run from the command line, use the following instructions:

First, ensure you have followed the guidelines in 4.1.2 Add a new Linux User Id.

LANSA provides a script, setenv.sh, that sets up the environment variables required for command line LANSA operations using the standard shell 'sh' or 'bash'. If you use other shell, e.g. csh, you will need to create you own script based on setenv.sh. This script must be loaded in a shell prompt (only once is required for the whole live time of the shell prompt) using the shell dot '.' command before running any LANSA command line in the same shell prompt.

For example, assuming you are using the standard shell 'bash' for Red Hat Enterprise Linux, the command line to load the script looks like:

./lansa/x lansa/bin/setenv.sh

Notice the space between the first dot and the following slash is part of the command line and cannot be omitted.

Alternatively, you may set the LANSAXROOT environment variable first:

LANSAXROOT=/lansa; export LANSAXROOT

. \$LANSAXROOT/x_lansa/bin/setenv.sh

If any errors are found during the script's execution, error messages will be issued and the scripts return code will be non-zero.

Optionally, you can change the user's login script to load setenv.sh whenever a shell prompt is started. That would be handy if you use LANSA command line frequently.

4.2 Modify LANSA Configuration Files and Environment Variables

To modify listener or communications options such as the TCP/IP port used, encryption, logging and tracing levels, the name and/or location of lroute.dat and trace files, refer to 4.2.1 Configure LANSA Communications.

To modify system and partition defaults for x_run arguments, refer to 4.2.2 Set up default x_run parameters.

4.2.1 Configure LANSA Communications

The *Linux Configuration* chapter in the *LANSA Communications Setup Guide* explains when and how to make these modifications.

If you need to change the port of the LANSA listener from the default of 4545, remember to modify the client's *Communication Identifier* field to use the new port number.

lroute.dat

The lroute.dat file shipped with the LANSA installation is just to get you started. To enable tracing or add host routes, etc. you need to replace or modify the lroute.dat file.

listen.dat

The listen.dat file installed with the LANSA installation is just to get you started. To enable encryption, logging and tracing, etc. you need to replace or modify the listen.dat file.

lansa.ini

On Windows, configuration items for LANSA Communications are maintained in the LANSA Communications Administrator and stored in the registry on Windows.

On Linux, these configuration items are stored in the lansa.ini file, which is located by default in the \$LANSAXROOT directory. The lansa.ini file uses the.ini file format used before there was a registry in Windows.

Note: Always stop the listener before modifying/replacing the lroute.dat, listen.dat or lansa.ini files, and then restart the listener.

4.2.2 Set up default x_run parameters

Although most x_run parameters are automatically received from the Client, it is often easier if standard parameters can be configured in one place. For example, a specific directory for RPTH=, or a DBID= setting for each partition.

An overview of the facilities LANSA provides to set the defaults is given below, but please refer to Permanently Specifying X_RUN Parameters in the *Technical Reference Guide* for details.

x_lansa.pro

The x_lansa.pro file allows default x_run parameters to be specified at the partition level.

You can also use an x_lansa.pro file to specify default x_run parameters for the whole LANSA installation. The system level x_lansa.pro file is created in the \$LANSAXROOT/x_lansa directory.

X RUN environment variable

The X_RUN environment variable allows default x_run parameters to be specified for LANSA sessions, i.e. incoming client requests or x_run executed from the command line.

For incoming client requests to inherit the X_RUN environment variable changes, the X_RUN environment variable must be changed in the root user's command shell and the listener has to be stopped and restarted in the same command shell.

If you need to change the X_RUN environment variable, you should always change it after calling the uservar.sh script which sets it to "ROOT:

\$LANSAXROOT". Use the following method to append default parameters to the X_RUN variable:

X_RUN="\$X_RUN"; export X_RUN

(Replace by the appropriate parameters.)

Precedence for default X_RUN parameters

If the same x_run parameter (e.g. DBID=) was specified on the incoming client request, in the X_RUN environment variable, in the system level x_lansa.pro and the partition level x_lansa.pro, which setting would be used?

Order of precedence for incoming **client requests**:

1. Parameters in the client request.

- 2. The X_RUN environment variable (inherited from the root user when the listener was started).
- 3. The partition level x_lansa.pro file.
- 4. The system level x_lansa.pro file

Order of precedence for command line execution of **x_run**:

- 1. Parameters on the command line.
- 2. The X_RUN environment variable (in the current shell when x_run is called).
- 3. The partition level x_lansa.pro file.
- 4. The system level x_lansa.pro file.

4.3 Start Multiple Listeners

In some circumstances it is necessary to have two separate LANSA listeners. This is usually the case when you have two separate installations of LANSA on the same machine; for example, one for testing and one for production. You may want to stop and start the listener in the testing environment to experiment with various settings, without affecting production users.

If you have more than one listener to support different LANSA installations, you should create completely different scripts to start and stop each listener. Please refer to 4.2 Modify LANSA Configuration Files and Environment Variables for further details.

Please refer to Running Multiple Listener Jobs in the *LANSA Communications Setup Guide* for further details.

IBM i Licenses: The license codes returned will contain this LANSA system's permanent licenses plus LANSA Integrator licenses where Integrator has been installed or upgraded using the IBM i installation process. Where LANSA Integrator has been installed separately, the Integrator licenses will be be returned by GET_LICENSE_STATUS.

4.4 Automate listener startup and shutdown

Your Linux system administrator can add the scripts created in Create listener startup and shutdown scripts into the startup and shutdown configuration for your system. Ensure that clients can connect as expected; it may be necessary to add additional environment variables into the scripts.

Please refer to 4.3 Start Multiple Listeners for details on handling multiple listeners.

4.5 Back up application data

When you back up your application data, you must always back up the set of .RRN files that is associated with that data. By default, the .RRN files are located in the partition directory, i.e. \$LANSAXROOT/x_lansa/x_ppp.

For more information on .RRN files, please refer to Handling Relative Record Numbers in the *LANSA Application Design Guide*.

5. Install LANSA Integrator on Linux

To use LANSA Integrator on Linux, you need to perform the following task:

5.1 Task: Install LANSA Integrator on Linux

You may also need to perform one or more of the following configuration and set up tasks:

5.2 Task: Configure the Java Services Manager on Linux

5.3 Task: Test Java Services Manager on Linux

5.4 Task: Specify Default Server for JSM Client

5.5 Task: Configure Environment Variables

5.1 Task: Install LANSA Integrator on Linux

Step 1. Prepare for LANSA Integrator Installation

- Optionally, request the appropriate LANSA licenses. Licenses are only required at execution time, so you can request and install licenses after the install. Refer to 5.2 Task: Configure the Java Services Manager on Linux for more information.
- Install a Java Runtime Environment (JRE) on your Linux machine. You can install Integrator without a JRE, but you will not be able to run it until a JRE is installed. Java 1.4 or greater is required.
- Optionally, install LANSA for Linux. You should do this before installing LANSA Integrator if you plan to use Integrator from LANSA functions on the same Linux machine. If you have done this, the following items in this step will already have been completed:
 - Optionally, create a new Linux group for the LANSA installation. This will allow you to restrict access to the LANSA files and directories to users who belong to a group.
 - Create a Linux user id to 'own' LANSA for Linux. This user id is referred to as the LANSA owner.
 - Create a directory for the LANSA installation. For example: /lansa. This directory is known as LANSAXROOT. A full install of LANSA Integrator takes around 11Mb of disk space (in addition to any other LANSA products you may have installed in the same location).
 - Modify the LANSA owner's login script to export the environment variable LANSAXROOT set to the file system created for LANSA. Set umask to 022 if you wish any Linux user to be able to access the LANSA files and directories. Set umask to 027 if you wish to restrict access to users who belong to the same group as the LANSA owner.
- Alternatively, you may use the LANSA for Linux Installer to do the full
 installation, which would install both LANSA for Linux and LANSA
 Integrator for Linux under the same installation directory. LANSA Integrator
 for Linux installed this way will use the LANSA for Linux in the same
 installation by default. Refer to the LANSA for Linux Installer Overview for
 more information.

Step 2. Run the Installer

As the root user, change directory to the install directory and execute the installation script to copy files to their correct locations.

For default installation, i.e. installing all LANSA Integrator for Linux components, the following command line will check if the provided options are valid:

Note that the path for the option --integrator-lansa-root is referring to the LANSA for Linux installation that this LANSA Integrator for Linux installation is going to use. If that LANSA for Linux installation and this LANSA Integrator for Linux installation are installed into two different directories, the path for the option

--integrator-lansa-root would be different to that for the option -I that is referring to where this LANSA Integrator for Linux installation is going to reside.

After the above command is run, nothing from LANSA Integrator for Linux will be installed or modified just yet. The command will report if the options are correct and make sense. If there is error, please adjust the command line according to the command output. If there is no error, you can then commit the installation by adding the option --commit to the command line. The command line becomes:

For customised installation, run the following command to see the full list of options for the installer:

```
./liinstall.py -h
```

• Once the install script has completed, please look at the output for any

problems that occurred during the install.

5.2 Task: Configure the Java Services Manager on Linux

After you have installed the LANSA Integrator, you must check that the default instance values are suitable.

Before You Begin

Before you begin to configure the JSM Server on Windows, you should be able to answer "yes" to the following questions:

- Have you installed the JSM Server software using the LANSA Linux install software?
- Do you have your new license codes for LANSA Integrator? If not, then request a license from your software vendor and supply the details specified in 5.6 Request and Record a LANSA Integrator License.
- Do you know the directory structure that will be used by the LANSA Integrator install?
- Do you know the IP address of the JSM Server?
- Do you have a JRE installed? You can install LANSA Integrator without a JRE, but you will not be able to run it until a JRE is installed. Java 1.4 or greater is required.
- Have you reviewed the Log files that are produced in the JSM instance trace?
 - These files may contain information regarding any updates required to property files.

JSMDirect and JSMProxy

If you have installed JSMDirect and JSMProxy, the \${INTEGRATOR_ROOT}/integrator/cgi-bin directory will contain three files: jsmdirect, jsmproxy and jsmadmin.

- jsmdirect and jsmproxy execute JSMDirect and JSMProxy when called via CGI.
- jsmadmin is used to view or change the JSMDirect and JSMProxy function registries (dc_w29.txt and dc_w30.txt) via Integrator Studio.

The file \${INTEGRATOR_ROOT}/integrator/apache/jsm.conf will be created containing all the Apache Web Server configuration required for LANSA Integrator for Linux. According to the Apache Web Server related options provided, the system Apache Web Server configuration file httpd.conf

will also be modified to include this file by default. So after you restarted the Apache Web Server, the CGI programs for LANSA Integrator for Linux will become ready to be activated from a web browser.

• You will need to check/update the values in **\${INTEGRATOR_ROOT}**/integrator/jsmdirect/jsmdirect.cfg Take particular note of the entries relating to the database.

Properties Files

If you have used the default installation directory, you will find the following directory structure installed on your server:

\${INTEGRATOR_ROOT}/integrator/instance/...

Within the instance directory are several directories, which contain files used to configure the JSM server. You should be aware of the contents of the following directories:

\$\{INTEGRATOR_ROOT\}\/integrator\/instance\/properties \$\{INTEGRATOR_ROOT\}\/integrator\/instance\/system

If you are using tracing, you will also need to locate this directory: \$\{\text{INTEGRATOR}_\text{ROOT}\}/\text{integrator/instance/trace}

5.3 Task: Test Java Services Manager on Linux

Four different test scenarios are provided to reflect the different ways that you can use LANSA Integrator on Linux. Select the scenario that matches how you intend to use Integrator.

- 5.3.1 JSM via JSMDirect
- 5.3.2 JSM from LANSA for IBM i or Windows
- 5.3.3 JSM from LANSA for Linux
- 5.3.4 JSM from a 3GL program

Once you have tested that your Integrator configuration works, continue to 5.3.5 Further Examples for more example material.

5.3.1 JSM via JSMDirect

This test matches the most common scenario, where you want to use the JSM via an HTTP connection. In this test, a web browser (for example, Netscape or Internet Explorer) is used to test the JSM setup. For the later tests listed in 5.3.5 Further Examples, you will also need to install the User Agent.

Before You Begin

Before you begin to test the JSM Server, check the following:

- Have you installed the JSM Server and JSMDirect on Linux?
- Have you installed the JSM Server licenses?
- Do you know the IP address of the JSM Server on Linux?
- Is there an HTTP server running on the Linux server, and is it using the generated jsm.conf?
- Have you installed LANSA for Linux?
- For the later examples, have you installed the User Agent on Windows, Linux (with at least Java 1.4)?

Step 1. Start JSM

Use the strjsm script to start a JSM instance.

If the JSM Server does not start, or stops after a few seconds, refer to Troubleshooting in the *Installing LANSA on Windows Guide*.

Step 2. Test JSMDirect

From a web browser, call the JSMDirect CGI program with no arguments. In most cases, this will be at http://linuxhost/cgi-bin/jsmdirect where linuxhost is the host name or IP address of your Linux server.

You should see a message like this:



Step 3. Review Example Code

For a function to test JSMDirect, refer to the JSMDirect Example in the *LANSA Integrator Guide*. To deploy this function on the Linux server, refer to the *Deploying LANSA Applications on Linux Guide* for details.

Make sure that you use the IP address or host name of the Linux JSM Server in your JSM_OPEN command. For example:

USE BUILTIN(JSM_OPEN) WITH_ARGS('LINUXHOST:4560') TO_GET(#JSMSTS #JSMMSG)

Step 4. Register Function with JSMDirect

Add the following line to the text file \${INTEGRATOR_ROOT}/integrator/jsmdirect/dc_w29.txt: TEST,*DEFAULT,JSMTEST,EXAMPL8,DEM,ENG

The example assumes that EXAMPL8 is the name of the example function in the JSMTEST process of the DEM partition.

Once this line has been added, test the function by opening http://linuxhost/cgi-bin/jsmdirect?test in your browser.

5.3.2 JSM from LANSA for IBM i or Windows

In this scenario, you will use LANSA functions on an IBM i or Windows machine to send commands to the JSM Server on Linux.

Before You Begin

Before you begin to test the JSM Server, check the following:

- Have you installed the JSM Server on a Linux machine? (JSMDirect is not necessary.)
- Have you installed the JSM Server licenses?
- Do you know the IP address of the JSM Server on the Linux machine?
- Have you installed LANSA on IBM i or Windows?

Step 1. Start JSM

Use the strjsm script to start a JSM instance.

If the JSM Server does not start, or stops after a few seconds, refer to Troubleshooting in the *Installing LANSA on Windows Guide*.

Note the Process Identifier used.

Step 2. Review Example Code

For code that you can use to test the Java Service Manager (JSM), refer to Appendix B. Client Application Examples in the *LANSA Integrator Guide*. A good starting point is Open/Close Example which tests that you have correctly started and configured the JSM.

Make sure that you use the IP address or host name of the Linux JSM Server in your JSM_OPEN command. For example:

USE BUILTIN(JSM_OPEN) WITH_ARGS('linuxhost:4560') TO_GET(#JSMSTS #JSMMSG)

5.3.3 JSM from LANSA for Linux

In this scenario, you will use server-side LANSA functions on Linux to send commands to the JSM Server on the same machine. You can either use a LANSA client on IBM i or Windows to initiate the function, or run the functions in batch mode.

Before You Begin

Before you begin to test the JSM Server, check the following:

- Have you installed the JSM Server on the Linux machine? (JSMDirect is not necessary.)
- Have you installed the JSM Server licenses?
- Have you installed LANSA on the Linux machine?
- Do you know the IP address of the JSM Server on the Linux machine?
- If you are planning to use a LANSA client on IBM i or Windows:
 - Have you installed LANSA on IBM i or Windows?
 - Is the LANSA listener running on Linux?

Step 1. Start JSM

Use the Strjsm script to start a JSM instance. Note that it will print the process ID of the new instance script.

If the JSM Server does not start, or stops after a few seconds, refer to Troubleshooting in the *Installing LANSA* on *Windows Guide*.

Step 2. Execute Example Function

If you are using an IBM i or Windows client, execute the client portion of the function. The LANSA User should be the same as the username to be used on the Linux server.

The example functions will ask you for some parameters:

- Server Name the name of the Linux server. Refer to the *LANSA Communications Setup Guide* for details about setting up communications with other servers.
- Password the password to be used on the Linux server.
- Arguments the X_RUN arguments to be used when executing the server function. This should include database details as a minimum; for testing, it should also indicate what level of tracing to use. See below for example

parameters.

Be sure that tracing is ON, as you will need it in the next step, *Check Tracing Files*.

If you are executing the functions in batch mode, you need to call x_run from the Linux command line with the appropriate parameters. Here is an example command that calls function EXAMPL1 in process JSMTEST, using a database called MY_DB, as user LANSAUSER:

x_run PROC=JSMTEST FUNC=EXAMPL1 DBID=MY_DB DBII=MY_DB DBUT=ORACLE DBIT=ORACLE DBUS=SYSTEM PSWD=MANAGER PART=DEM LANG=ENG USER=LANSAUSER ITRO=Y ITRL=4

Refer to Override X_RUN arguments inherited from the Client in the *Deploying LANSA Applications on Linux Guide* for details.

Step 3. Check Trace Files

For functions run in batch mode, the best way to check whether they have run successfully is to check for new trace files. Trace files will be present if you left tracing on in the preceding step, *Execute Example Function*.

To find the trace files, locate the directory inside

\${INTEGRATOR_ROOT}/integrator/instance/trace matching the JSM instance's process ID (refer to Step 1. Start JSM). If the function was executed successfully, this directory will contain a subdirectory called CLIENT0000001 (unless you have turned tracing off). If either directory does not exist, refer to Troubleshooting.

5.3.4 JSM from a 3GL program

This test matches the scenario where you will call JSM commands from your own program written on Linux in a 3rd Generation Language (3GL) such as C.

Before You Begin

Before you begin to test the JSM Server, check the following:

- Have you installed the JSM Server on the Linux machine? (JSMDirect is not necessary.)
- Have you installed the JSM Server licenses?
- Do you know the IP address of the JSM Server on the Linux machine?
- Do you have an appropriate 3GL compiler set up on the Linux machine?

Step 1. Start JSM

Use the strjsm script to start a JSM instance. Note that it will print the process ID of the new instance script.

If the JSM Server does not start, or stops after a few seconds, refer to Troubleshooting in the *Installing LANSA* on *Windows Guide*.

Step 2. Locate the JSM Shared Library

The JSM shared library, libx_jsm.so, contains the functions you will use to communicate with the JSM. This file needs to be available when you compile and run your 3GL program. For C programs, x_jsm.h contains the API for these functions.

For more information about using the JSM shared library functions, refer to Java Service Manager Clients in the *LANSA Integrator Guide*.

Step 3. Compile and Run a 3GL Program

Some example programs written in C are provided in C examples in the *LANSA Integrator Guide*. A good starting point is Open/Close Examples in the *LANSA Integrator Guide*. This example tests that you have correctly started and configured the JSM. The example provides some debugging output to show whether the connection to the JSM was successful.

If you get an error message of the form "JSMMSG nnn", where nnn is a 3-digit number, you can look it up in C Client Error Messages in the LANSA Integrator Guide.

The example program allows you to enter an IP address or host name as a

command line argument. When writing your own 3GL program, make sure that you use the IP address or host name of the Linux JSM Server in your JSMOPEN function call. Alternatively, since the function will run on the same machine as the JSM, you can use "LOCALHOST" to refer to the local machine. For example:

JSMOPEN ("LOCALHOST:4560", status, response);

5.3.5 Further Examples

Review the Tutorials in the *LANSA Integrator Guide* for a set of introductory exercises.

You should also refer to the SET materials on the LANSA Software DVD for examples of LANSA Integrator use:

- Using the JSM to send email
- Using the JSM to manage FTP sessions for file transfers
- Accessing the NASDAQ Web service stock quote inquiry
- Using the JSM for the client side of a XML transaction (place an order)
- Using the JSM for the client side of a B2B XML query (query an order's status)
- Using the JSM (JSMDirect) for the server side of a B2B XML transaction (handle order placement)
- Using the JSM (JSMDirect) for the server side of a B2B XML query (handle order status query)
- Using the JSM to perform remote procedure calls (RPCs) to LANSA functions on other platforms.

5.4 Task: Specify Default Server for JSM Client

During a Visual LANSA install, a default file (jsmcltdta.txt) is inserted into the LANSA Execution Environment's path (\${INTEGRATOR_ROOT}/x_lansa).

Below is the jsmcltdta.txt file's content when it is first installed?

localhost:4560 is the installed value for DefaultServer

If you are connecting to a different server and/or using

a different port, please change the setting below to reflect

the server and/or port, ie,

DefaultServer=MyHost:4560

DefaultServer=localhost:4560

ExcludePrefix=JSM

Please change these two entries as required:

Keyword Comment

DefaultServer Defines the default location of the JSM Server. If JSM_OPEN

is called without arguments then the default server location as

specified in jsmcltdta.txt will be used. Example:

DefaultServer=localhost:4560

ExcludePrefix Defines which fields will be excluded from being exchanged

with JSM Server. Example:

ExcludePrefix=JSM

5.5 Task: Configure Environment Variables

- 5.5.1 Set the PATH Variable
- 5.5.2 JSMDirect Execution Environment

5.5.1 Set the PATH Variable

The PATH environment variable, which tells the system where to find programs to be run from the command line, should include entries pointing to LANSA Integrator programs for each user who will need to use them.

All LANSA Integrator programs except for JSMDirect and JSMProxy should be run from the \${INTEGRATOR_ROOT}/integrator/jsmadmin directory:

- strjsm starts the JSM Server. This should either be run as a background task or in a separate shell, and is usually run by the LANSA owner.
- useragent Integrator User Agent.

Each of these scripts also needs to be able to find the java command. This usually means including the bin directory of your Java installation in the PATH variable, unless it has already been installed in, for example, /usr/bin.

The login script of each LANSA user should be modified to include the appropriate entries, for example:

JAVA_HOME=...
PATH=\$PATH:\${INTEGRATOR_ROOT}/integrator/jsmadmin:\$JAVA_HOMI
export JAVA HOME PATH

5.5.2 JSMDirect Execution Environment

The file \${INTEGRATOR_ROOT}/integrator/jsmdirect/jsmdirect.cfg contains settings which are used when you run JSMDirect. The most important entry is LANSAXROOT. This entry points to the root of the LANSA install which will execute the X_RUN request.

You may need to modify this file if:

• You move your Visual LANSA installation, or Visual LANSA is installed in a different place to LANSA Integrator.

5.6 Request and Record a LANSA Integrator License

To obtain an Integrator license, you must submit a written request to your product vendor giving them the following details:

- Machine CPU serial number and model type.
 Run the LANSA utility x_cpu at the command line to display this information.
 - The x_cpu utility is installed automatically with LANSA during an Linux installation. If you don't have a LANSA installation yet, request a copy of the x_cpu utility from your product vendor.
- Type of license required.
- If a seat license, the number of seats.
- Whether a TEMPORARY or PERMANENT license is required.
- If a temporary license is requested, the expiry date required.

Record the LANSA Integrator license as described in Licensing on the LANSA web site.

6. Install LANSA for the Web Apache Module

If you want to use a web server on Linux then you will need to install the LANSA for the Web Apache Module. Following are instructions for this installation.

Supported Platforms:

• Red Hat Enterprise Linux Server release 6.X (x86_64)

Supported Web Server:

• Apache Version: 2.2.x

Note: Apache worker MPMLANSA for the Web Apache Module supports both Apache MPMs prefork (single-threaded and usually the default) and worker (multi-threaded). However, LANSA/Open connection pooling capability is only available for the worker MPM. So it is highly recommended that you use the worker MPM. To use the worker MPM in Red Hat Enterprise Linux Server, change the setting HTTPD to refer to the Apache daemon named httpd.worker instead of the default httpd, by editing the following file:

/etc/sysconfig/httpd

Also see

- 6.1 Installation Steps
- 6.2 Configure the LANSA for the Web Apache Module
- 6.3 Verify the Installation

6.1 Installation Steps

- 1. Run a shell as root.
- 2. Copy the installation package for the target platform from the LANSA CD to a temporary directory, e.g. /tmp.

The installation package is for these supported platforms:

• Red Hat EL6 (x86_64): wpi13200_lnx64.tgz

This file will be referred as wpi13200 in the following steps.

3. Enter the following commands in the shell:

cd/tmp

gunzip -c wpi13200_lnx64 | tar xvf -

A directory named wpi13200 will be created with all the installation material extracted.

4. Change current directory to wpi13200 with the following command:

cd wpi13100_lnx64

5. Execute the installer with the following command to get an idea of the parameters required and options that are available:

./wpiinstall.py --help

6. Notice that installation will NOT be commenced until option **--commit** is given, so it is perfectly safe to try various combinations of parameters and options and then check the report from the installer. The installer will report any error found or list the parameters to be used if there is no error. Once all the parameters are verified and fit the system's requirement, installation can be commenced by re-running the installer with the required parameters and option **--commit** added.

Some examples:

a. Red Hat Linux EL6 with default Apache setup:

Enter the following command to view installation settings for installing the Apache module to the directory /opt/lansaweb:

./wpiinstall.py -I /opt/lansaweb

To install, enter this command:

./wpiinstall.py -I /opt/lansaweb --commit

b. Red Hat Linux EL6 with custom Apache installation in the directory /opt/httpd-2.2:

./wpiinstall.py -I /opt/lansaweb -a /opt/httpd-2.2 --commit

c. Using a custom Apache configuration file:

./wpiinstall.py -I /opt/lansaweb -a /usr/local/apache2 \
-C /usr/local/etc/myhttpd.conf --commit

7. Read the installation log (on screen) for further information about the installation. The log can be saved for later perusal by running the installer with a command like this:

./wpiinstall.py {options} 2>&1 | tee wpiinstall.log

8. For Red Hat Enterprise Linux Server, for the additional manual setup required, if SELinux is installed and enabled, read the file:

{install_root}/webserver/apache/selinux.notes

6.2 Configure the LANSA for the Web Apache Module

- 1. Create a Communications Route Table (lroute.dat) with the LANSA Communications Administrator on Windows. For details, refer to Configuring the Communications Route Table in the *LANSA Communications Setup Guide*.
 - Copy this route table from Windows to {install_root} on the Linux system. Alternatively, for advanced users, you may edit the file with a text editor.
- 2. Create a Web Server configuration file (l4w3serv.cfg) with LANSA for the Web Administrator as described in 6.2.1 Configure the LANSA for the Web Configuration file (l4w3serv.cfg)

Copy this configuration file from Windows to {install_root}/webserver/apache on the Linux system and ensure that the file name is all lower case and the file is readable by the apache daemon.

6.2.1 Configure the LANSA for the Web Configuration file (l4w3serv.cfg)

The Servlet configuration file contains the connection information used by the Servlet to connect the Web Server to the Data/Application Server. The configuration file contains the connection information to connect the Web Server to the Data/Application Server. This file is created and maintained using the LANSA for the Web Administrator (Local Configuration) and must be named 14w3serv.cfg. (Note that the file name must all be in lower case).

The following example describes a simple setup and does not make use of options such as Authentication or HTTP Header variables.

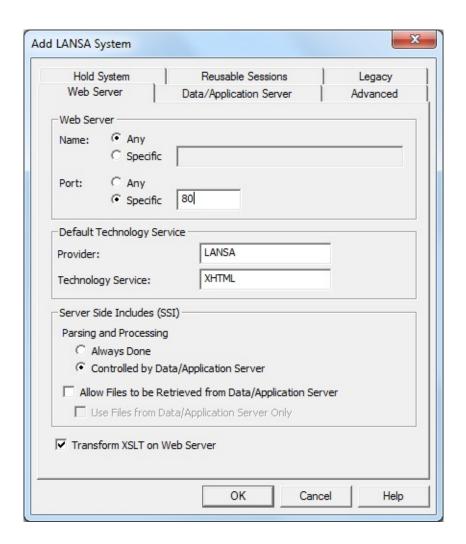
Follow these steps to create a simple Servlet configuration file.

Define the Web Server:

- 1. Start the LANSA for the Web Administrator by opening the LANSA folder from the LANSA icon on your desktop, selecting the Setting and Administration folder and choosing the Web Administrator from the list.
- 2. From the *Options* menu select *Local Configuration* and choose *New* to create a new Servlet configuration file
- 3. From the *Tools* menu select *Web Server* and verify that the default settings displayed on the *File Location* tab and the *Trace* tab are valid for your installation.

Define the connection parameters to the Data/Application server:

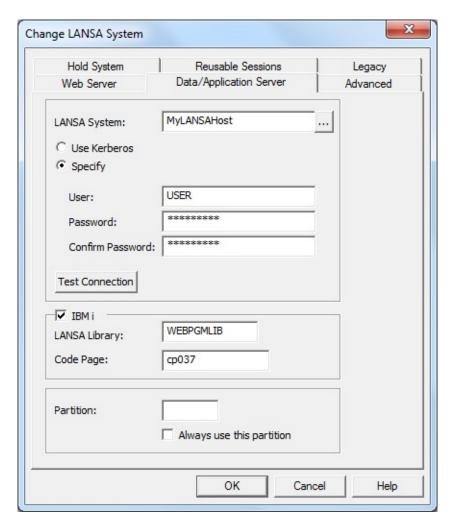
- 1. Select the *Tools* menu, choose the *Maintain Systems* sub-menu to display the *Maintain Systems* (local) dialog box.
- 2. In this dialog, press the *Add* button to display the *Add LANSA System* dialog.
- 3. Select the *Web Server* tab and specify the parameters.



In this sample installation, Any is selected as the *Web Server Name* and the Port number is specified as '80' (the default port for HTTP).

If you enter a *Web Server Name*, be aware that due to restrictions on some DNS servers, it is necessary to enter it in lower case.

4. Select the *Data/Application Server* tab and specify the parameters.



In this sample installation, 'MyLANSAHost' is entered as the *LANSA System*. This parameter must the same as the *Partner LU Name* specified in the Communications Route Table (lroute.dat) file.

'cp037' is entered as the *Code Page* if the Data/Application Server is an IBM i Data/Application Server.

- 5. The default settings are used for the remaining tabs.
- 6. Save the Servlet configuration file with the name l4w3serv.cfg. Make sure that you save the file with the correct name and that the file name is in lowercase.

6.3 Verify the Installation

1. Execute the following command to start the Apache Web server:

{install_root}/webserver/apache/apache_start

2. Load the following URL in a Web Browser:

http://{your_webserver}/cgi-bin/about

3. To test if the assigned LANSA Web Server can be connected:

http://{your_webserver}/cgi-bin/lansaweb?about

4. To test if the assigned LANSA Data/Application Server can be connected:

http://{your_webserver}/cgi-bin/lansaweb?
wam=xabout&webrtn=show&part={your_partition}

5. Execute the following command to stop the Apache Web server:

{install_root}/webserver/apache/apache_stop

Appendix A. Help Text

Help Text for LANSA for Linux Installer
Help Text LANSA Integrator for Linux Installer
Help Text for LANSA for the Web Apache Module for Linux Installer

Help Text for LANSA for Linux Installer

You can get the full help text of LANSA for Linux Installer at a shell prompt by running the command:

./vlinstall.py -h

The full help text of LANSA for Linux Installer is reproduced here for your reference:

```
program: vlinstall.py
version: XX.X.X
description: Visual LANSA for Linux XX.X.X Installer
usage: ./vlinstall.py [options]
installation types:
  --full-install
     Install everything available in the installation package
  --visual-lansa-only
    Install Visual LANSA Runtime only (default)
  --web-server-only
    Install LANSA for the Web Plug-in (Apache Module) only
  --connect-only
     Install LANSA Connect only
  --integrator-only
     Install LANSA Integrator only
  --integrator-options {jsmserver,jsmproxy,useragent,rfi}
    LANSA Integrator options to install
     allow any combinations of options
     (default to install all LANSA Integrator options)
options:
```

- -I {dir}, --install-root {dir} installation root directory
- -O {uid}, --install-owner {uid} installation owner user id (see "need root" below)
- -G {gid}, --install-group {gid} installation owner group id (see "need root" below)

```
-D {dsn}, --dsn {dsn}
  ODBC data source name
-u {dbuid}, --db-user {dbuid}
  ODBC data source user id
-p {dbpw}, --db-password {dbpw}
  ODBC data source password
--create-dsn
  create/fix ODBC data source name (see "need root" below)
  required only if ODBC data source is *not* functional
--oracle-server {server}
  Oracle server host name for --create-dsn
--oracle-sid {sid}
  Oracle SID for --create-dsn
--install-oracle-client
  install Oracle Instant Client 11 if required
  (see "need root" below)
-n, --no-init
  do not initialize database
-l {port}, --listener-port {port}
  listener port (default to 4545)
-L {language}, --language {language}
  installation language
  (default to "ENG")
-a {dir}, --apache-root {dir}
  specify the Apache server root directory
  (default to "/etc/httpd")
-m {dir}, --apache-modules-dir {dir}
  specify the Apache server modules directory
  if omitted, will derive the path from --apache-root option
  (default to "/etc/httpd/modules")
-C {file}, --httpd-conf {file}
  specify the path of the Apache configuration file httpd.conf
  if omitted, will derive the path from --apache-root option
  (default to "/etc/httpd/conf/httpd.conf")
-H {file}, --httpd {file}
  specify the path of the Apache daemon httpd
  if omitted, will search for the file according to the --apache-root
  option and the standard places
-A {file}, --apachectl {file}
```

specify the path of the Apache control program apachectl if omitted, will search for the file according to the --apache-root option and the standard places --apache-user {user} specify the user name which the Apache daemon runs as if omitted, will try to determine that from the Apache configuration file httpd.conf --apache-group {group} specify the user group which the Apache daemon runs as if omitted, will try to determine that from the Apache configuration file httpd.conf --integrator-lansa-root {dir} Visual LANSA root directory for LANSA Integrator must be provided for LANSA Integrator (jsmserver option) only installation (default to installation root directory for full install) --integrator-cgi-prefix {cgipath} LANSA Integrator CGI URL prefix (default to "cgi-bin") -c, --commit commit installation or other action -i, --install install using provided options (default action) -r, --reinstall re-install to overwrite existing installation -F, --forced-reinstall forced re-install to overwrite existing *NEWER* installation -V, --verbose verbose mode -v, --version print version -h, --help print this help action selection (instead of install): --check-install check installation --fix-install-ownerships

fix installation ownerships (see "need root" below)

need options -O/-G
--fix-install-permissions
fix installation file permissions (see "may need root" below)
--db-init
initialize database only
need options -D/-u/-p
--system-init

"need root" notes:

This installer *must* be run with root privileges for the options to be effective.

"may need root" notes:

perform system init only need options -D/-u/-p

This installer *may* need to be run with root privileges for the options to be effective.

Help Text LANSA Integrator for Linux Installer

You can get the full help text of LANSA Integrator for Linux Installer at a shell prompt by running the command:

```
./liinstall.py -h
```

The full help text of LANSA Integrator for Linux Installer is reproduced here for your reference:

```
program: liinstall.py
version: XX.X.X
description: LANSA Integrator for Linux XX.X.X Installer
usage: ./liinstall.py [options]
options:
  --integrator-options={jsmserver,jsmproxy,useragent,rfi}
     LANSA Integrator options to install
     allow any combinations of options
     (default to install all LANSA Integrator options)
  -I {path}, --install-root={path}
     installation root directory
  -O {uid}, --install-owner={uid}
     installation owner user id (see "need root" below)
  -G {gid}, --install-group={gid}
     installation owner group id (see "need root" below)
  -a {dir}, --apache-root {dir}
     specify the Apache server root directory
     (default to "/etc/httpd")
  -m {dir}, --apache-modules-dir {dir}
     specify the Apache server modules directory
     if omitted, will derive the path from --apache-root option
     (default to "/etc/httpd/modules")
  -C {file}, --httpd-conf {file}
     specify the path of the Apache configuration file httpd.conf
     if omitted, will derive the path from --apache-root option
     (default to "/etc/httpd/conf/httpd.conf")
  -H {file}, --httpd {file}
```

```
specify the path of the Apache daemon httpd
    if omitted, will search for the file according to the --apache-root
     option and the standard places
  -A {file}, --apachectl {file}
     specify the path of the Apache control program apachectl
     if omitted, will search for the file according to the --apache-root
     option and the standard places
  --apache-user {user}
    specify the user name which the Apache daemon runs as
     if omitted, will try to determine that from the Apache configuration
     file httpd.conf
  --apache-group {group}
    specify the user group which the Apache daemon runs as
     if omitted, will try to determine that from the Apache configuration
     file httpd.conf
  --integrator-lansa-root {dir}
     Visual LANSA root directory for LANSA Integrator
     must be provided for jsmserver option
  --integrator-cgi-prefix {cgipath}
     LANSA Integrator CGI URL prefix
    (default to "cgi-bin")
  -c, --commit
     commit installation or other action
  -i, --install
     install using provided options (default action)
  -r, --reinstall
    re-install to overwrite existing installation
  -F, --forced-reinstall
     forced re-install to overwrite existing *NEWER* installation
  -V, --verbose
     verbose mode
  -v, --version
     print version
  -h, --help
     print this help
action selection (instead of install):
  --check-install
```

check installation

- --fix-install-ownerships
 fix installation ownerships (see "need root" below)
 need options -O/-G
- --fix-install-permissions fix installation file permissions (see "may need root" below)

"need root" notes:

This installer *must* be run with root privileges for the options to be effective.

"may need root" notes:

This installer *may* need to be run with root privileges for the options to be effective.

Help Text for LANSA for the Web Apache Module for Linux Installer

You can get the full help text of LANSA for the Web Apache Module for Linux Installer at a shell prompt by running the command:

```
./wpiinstall.py -h
```

The full help text of LANSA for the Web Apache Module for Linux Installer is reproduced here for your reference:

```
program: wpiinstall.py
version: XX.X.X
description: LANSA for the Web Plug-
in (Apache Module) for Linux XX.X.X Installer
usage: ./wpiinstall.py [options]
options:
  -I {path}, --install-root={path}
     installation root directory
  -O {uid}, --install-owner={uid}
     installation owner user id (see "need root" below)
  -G {gid}, --install-group={gid}
     installation owner group id (see "need root" below)
  -a {dir}, --apache-root {dir}
     specify the Apache server root directory
     (default to "/etc/httpd")
  -m {dir}, --apache-modules-dir {dir}
     specify the Apache server modules directory
     if omitted, will derive the path from --apache-root option
     (default to "/etc/httpd/modules")
  -C {file}, --httpd-conf {file}
     specify the path of the Apache configuration file httpd.conf
     if omitted, will derive the path from --apache-root option
     (default to "/etc/httpd/conf/httpd.conf")
  -H {file}, --httpd {file}
     specify the path of the Apache daemon httpd
```

if omitted, will search for the file according to the --apache-root option and the standard places

-A {file}, --apachectl {file}

specify the path of the Apache control program apachectl if omitted, will search for the file according to the --apache-root option and the standard places

--apache-user {user}

specify the user name which the Apache daemon runs as if omitted, will try to determine that from the Apache configuration file httpd.conf

--apache-group {group}

specify the user group which the Apache daemon runs as if omitted, will try to determine that from the Apache configuration file httpd.conf

-c, --commit

commit installation or other action

-i, --install

install using provided options (default action)

-r, --reinstall

re-install to overwrite existing installation

-F, --forced-reinstall

forced re-install to overwrite existing *NEWER* installation

-V, --verbose

verbose mode

-v, --version

print version

-h, --help

print this help

action selection (instead of install):

--check-install

check installation

--fix-install-ownerships

fix installation ownerships (see "need root" below)

need options -O/-G

--fix-install-permissions

fix installation file permissions (see "may need root" below)

[&]quot;need root" notes:

This installer *must* be run with root privileges for the options to be effective.

"may need root" notes:

This installer *may* need to be run with root privileges for the options to be effective.