

Release Notes 2.0e 29/03/2001

Table of Contents

- L*i*SA overview
- What is new in L*i*SA - Release 2.0e release
- Welcome
- New features
- New enhancements
- Things you need to know
- CD structure
- Installation
- Migration
- Limits
- Troubleshooting

L*i*SA overview

L*i*SA stands for Lotus i-net Solution Architecture. It's a "solution" architecture because it is designed to re-use basic application components, so developers focus on adding value ... not on re-creating the basics. And "i-net" means the applications you develop can be used by Lotus Notes clients and simultaneously by a standard web browser over the Internet, intranet or extranet.

Suddenly your Lotus Notes investment is ready for i-net and e-business operation.

A consistent user interface means the training requirement for new applications is reduced. Ongoing changes to your "business basics" like people, organisations and processes can be cascaded throughout your L*i*SA applications.

Five key components

The 'foundation' components are the smart way to share information about your business between applications:

· L*i*SA Organization / Address
· L*i*SA Glossary
· L*i*SA Configure

The 'usability' components ensure that L*i*SA applications are easy to use and have a consistent interface:

· L*i*SA Help
· L*i*SA Corporate Look&Feel

What is new in L*i*SA - Release 2.0e

Welcome

L*i*SA release 2.0e is a publicly released maintenance version available on CD. It is identical with the first German 2.0 Release and includes few minor changes with regard to the previous Englisg 2.0 Release. Especially the LiSAOrganisation has been improved.

L*i*SA-Organization implements a job-based organisational database for a wide variety of applications, in particular for those with document rights, access control or workflow. In addition, L*i*SA provides the user with a reference source covering all aspects of the organisational structure.

Access to L*i*SA-Organization is provided by various standardised interfaces, which are laid out in such a way that they offer direct benefits to database designers while suitably encapsulating the data. The application achieves an extensive level of independence from the organisational database implementation.

The use of generated Domino Directory groups means that it is possible to implement a job reference which allows a delegate rule. Consistent use of the Domino Directory group functions prevents runtime and replication problems because no changes to the document are necessary.

L*i*SA-Address offers a comprehensive address management concept for end users, programmers and all individuals who wish to manage addresses efficiently. The L*i*SA-Address database combines all of the modules of the L*i*SA-Address concept and is at the same time a sample implementation. Interfaces for the standardisation of applications are available and can be implemented in business applications. All address pools that are available in an infrastructure can be detected and managed by L*i* SA-Configure.

L*i*SA-Glossary provides all of the necessary Lotus Notes and Domino design elements so that it can easily be integrated into other Lotus Notes and Lotus Domino applications and therefore access one or more central terminology databases. This L*i*SA component is a building block for several other L*i*SA components, including L*i*SA-Help and L*i*SA-Address.

L*i*SA-Configure is the central point for defining database references and application parameters on various levels. It "glues" applications together into one logical configuration. Application designers can easily access the L*i*SA-Configure data through encapsulated, powerful functions that keep the configuration logic outside of the applications.

In addition to providing a standardised, yet extensively configurable template for creating online manuals, **L*i*SA-Help** also provides central, context-sensitive Help for all Lotus Notes and Domino applications.

L*i*SA-Corporate Look&Feel takes away a part of the workload of the IT department by providing them with a tool to implement and manage the standard graphical functionality. Also, the ongoing improvements in standard components reduces the testing time so that the end users get better quality much quicker. This allows the IT department to spend more time on functional improvements or new technologies.

Because of the standard components and user interface, the end users will work in an environment that is familiar so training cost are reduced. By keeping the quality of the components up to date, the user acceptance can be significantly improved

Next to these five key components, other modules are included to ease the development. These are explained in further below.

L*i*SA 2.0 now supports the following server platforms

- Win 32 / NT

- OS/2,
- AIX,
- HP-Unix,
- Sun Solaris,
- S/390,
- AS/400

The platform Win 16 has been discontinued.
(Other product and company names mentioned herein may be the trademarks or registered trademarks of their respective owners.)

New features

- **L*i*SA-Configure**
  - new alias on "replica" level
  - full administration and editing features for browser clients
- **L*i*SA-Help**
  - truly context-sensitive help for browser applications using multiple frames
  - help documents readable in browser clients
- **L*i*SA-Address**
  - LotusScript classes for background processing of address documents
  - search agent serving the results as XML document
  - full editing features for browser clients
- **L*i*SA-Organization**
  - LotusScript classes for background processing of unit documents
  - dynamic traits
  - relational traits
  - all documents are readable in browser clients
- **L*i*SA-Glossary**
  - reference selection dialog now supports subtrees of the hierarchical structure
  - improved status handling ("reject", "reuse"...) including requester notification
  - full administration and editing features for browser client
- **L*i*SA-MessageHandler**
  - easy creation, test, and translation of message objects
  - generation of LotusScript message classes, JavaScript classes and Java resources that provide a wide range of functionality like summary messages, HTML creation, display functionality for LotusScript/JavaScript/Java and programmatic access even by @Commands.
- **L*i*SA-DesignTracker**
  - support for all R5 design elements
- **L*i*SA-LSDoc**
  - documentation generator to extract in-line documentation from LotusScript files and to generate an HTML documentation including index, table of contents and hyperlinks
- **L*i*SA-Version Control Tool**
  - VCT module fully supports R5 features
- **L*i*SA-UDF**
  - UDF.ini and UDFDebug.out reside in NotesData directory instead of the operating system's default directory
  - Notes.ini parameter LSACfgLink allows to define what database to use as L*i*SA-Configure; parameter format "Svr!!Path"
  - improved cashing to reduce network load
  - VCT module fully supports R5 features
  - @DesignAliasByPath reflecting the new meaning behind the alias
  - @GetMessage for access of MessageHandler generated LotusScript message libraries
  - @GetSystemTime : gives you a time value in milliseconds.

- For the programmer it is most important that the interfaces have been maintained and carefully extended. It has turned out that the serialized textual format of L*i*SA can be easily converted to

XML. Whenever meaningful, XML is used between backend services and applets. Many new Java applets have been programmed, which are embedded by Java script and LiveConnect. The calling interfaces greatly benefits from the serialized textual format, as structures cannot be easily exchanged between the applets and JavaScript.

- A complete new way of handling errors has been implemented into the L*i*SA modules. This makes it possible to share functionality between functions used in the Notes Client events and the backend agents running to support browser client features. The messages/errors are defined in a separate module, named L*i*SA-MessageHandler, that allows to generate message classes for LotusScript and JavaScript as well as Java resources. Access to messages in @Formula language is provided by new UDF functionality.

The CD also contains the backend-classes for L*i*SA-Organization and L*i*SA-Address as a separately charged item.

New enhancements

- **L*i*SA-Configure**
  - full editable in browser client
  - L*i*SA-navigation structure for browser clients
  - bug fix: the display of parameter lists is cut off if it exceeds the Notes memory limits
    "EditorParagraph Justification Pool is full"
    "Field is too large (32K) or View's column & selection formulas are too large"

- **L*i*SA-Help**
  - full readable in browser client
  - L*i*SA-navigation structure for browser clients
  - context sensitivity in browsers. The help takes into account multiple frames.
  - cookie based personal profiles.
  - The bug on closing the LiSA-Help has been fixed.

- **L*i*SA-Address**
  - full editable in browser client
  - L*i*SA-navigation structure for browser clients
  - dialog boxes for browser clients using combination of browser windows and Java applets
  - use of L*i*SA-MessageHandler functionality in LotusScript classes
  - new combined event model for Notes and browser clients to support use of same LotusScript classes
  - new address dialogs for Notes clients to avoid an OS/2 display issue
  - new set of backend classes for processing and import of documents in L*i*SA-Address

- **L*i*SA-Organization**
  - full readable in browser client
  - L*i*SA-navigation structure for browser clients
  - when saving organization units customer defined delimiters are considered for validation
  - bug fix: members dialog showed "no entries" when there were too many of them
  - entries in the members dialogs (also for surrogation rules and potential member types) are all sorted now
  - new address dialogues to avoid an OS/2 display issue
  - inconsistency view is also updated and shows following errors:
    - members not according to potential member type definition in type definition
    - recursions
    - missing Domino Directory names (SystemID)
    - unassigned addresses
  - improved icon selection for meta-types

- two new parameters in the L*i*SA-Configure to show hierarchy paths and competency paths only to certain roles: "LSAOrgShowHierarchy" and "LSAOrgShowTraits"
- address database views "PeopleGroupsFlat" and "PeopleGroupsHier" are included for selection in the Notes address box. Please include L*i*SA-Organization into File->Preferences->User Preferences->Mail and News->local address books
- field "LSAOrgIntTraitSearch" now has contents as described in the documentation
- improved display of hierarchy and traits paths in person documents
- no language document and no agent to create a language document are necessary anymore.
- In view Type Definitions/Traits the button to create new types has been corrected so that it works the same as the buton in Type Definitions/Groups. It is now possible to to create formula driven and relational traits.
- If you create a new trait in the Meta-Modell you may now use that to derive traits of this type and associate members.
- The trait view did not show cascaded traits correctly. The look of the view has been improved. The functionality is not impacted.
- improved database update:
  - is now triggered by response documents to the documents that were changed. This reduces the risk of replication conflicts.
  - no red arrows anymore for update request indication. View "Administration/UpdateResponses" shows update request documents instead. All update request documents can be viewed and all necessary information about the change(s) in the database are shown and you can open the changed document from it
  - better understandable log entry for update if a child unit was assigned to a parent unit
  - bug fixes
    when a unit was changed after its members had been deleted
    in type replacement after a type definition was removed (LSAOrgReference and LSAOrgMembers)
    SystemID is set empty if the NotesGroupType of a unit was set to "none"
    Default hierarchy path is also updated if a unit is removed as a member or another unit
    change of type definition also evaluated in the members field of a unit
    a log entry is generated if errors occur when documents are saved
  - The Agent SAOrgcNABUpdate aborted with a message "objectvariable not set". This issue had it's reason in an invalid configuration. The Error handling has been improved.
- improved Domino Directory update for groups; now includes calculation of surrogation
  - Domino Directory update of groups by alias works the same as for groups with their SystemID (NotesName)
  - surrogation update works the same for alias names as for the NotesName of the groups
  - Domino Directory update agent for group update starts internal database update
  - calculation of surrogates works correctly now also for several times and time ranges in the UnavailabilityTimes - field
  - calculation of surrogates works correctly also for surrogates with numeric middle initials
  - surrogates are always deleted from groups in Domino Directory after surrogation period ended
- improved Domino Directory update for persons
  - Person Update agent only removes documents if they were imported to the L*i* SA-Organization from the Domino Directory (field NABDocPointer is available)
  - person documents are properly removed after person update
  - recertification of persons is updated to the SystemID in the L*i*SA-Organization
- Interface classes
  - new unit can be initialized without display name
  - new NotesType "Interface" is equivalent to "Choose when created" option in type definitions; can be set by the interface class
  - change of the NotesType for a person unit with NotesType "Interface" works correctly from the script interface
  - Address subform is now corrected concerning it's Interface. In the previous version it could have happened, that units were showing up in the address interface erroneously. .
- Demo- Data for relational and dynamic traits are now corrected

- **L*i*SA-Glossary**
  - full editable in browser client
  - L*i*SA-navigation structure for browser clients
  - improved input validation avoids corruption of hierarchy data
  - LSADbOption "NONEWGLOS" to prevent the request of new Keywords in "see also" dialog
  - "see also" dialog allows to display a subtree of the glossary
  - improved synchronization during update, e.g. If a father document has been deleted and is reintroduced with the same name, all children in the subtree will be linked to this newly created document
  - requests can be relinked to existing keywords
  - notification in case of keyword requests, reuse, deletions

- **L*i*SA-MessageHandler**
  - standardized way of message handling
  - integrated test facility in Notes Client and browser
  - translation with Domino Global Workbench supported
  - two layer concept for general errors and module errors
  - generation of LotusScript and JavaScript Classes
  - generation of Java resources
  - access to messages by UDF-command @GetMessage
  - summarising of multiple error messages
  - wrapping of system errors supported
  - HTML generation for direct use of messages with browsers

- **L*i*SA-DesignTracker**
  - support of Domino Release 5 design elements
  - refresh functionality
  - deletion of projects

- **LSDoc**
  - reading in-line documentation from exported LotusScript library files (.lss-files)
  - generation XML documents representing the structure of the corresponding library
  - XSL-transformation, which is adoptable to customer needs, to generate HTML documentation readable in browser clients
  - predefined set of documentation tags to structure documentation (close to ones of the JavaDoc tool from SUN Microsystems)
  - creation of index and table of contents documents

- **L*i*SA-UDF**
  - **Enhancements**
    - Main-Module:
      The file udf.ini is now located in the Lotus Notes data directory. In prior versions the location depends on the operating system (e.g. NT : Windows directory, UNIX : current directory).
      The file udfdebug.out is now generated in the Lotus Notes data directory. In prior versions the location depended on the operating system.
      The UDF trace shows now the version and revision information of the UDF Main-Module as well as the loaded modules.
    - L*i*SA-Configure (CFG) - Module :
      With the parameter "LSACfgLink" in NOTES.INI, you can bypass the search for the L*i* SA-Configure database. If the parameter is present, the path or the server and path must point to a valid L*i*SA-Configure database.
      The results from the L*i*SA-Configure database are cached. This results in less network traffic and better performance.
      If the server component of the context equals the current machine, the L*i*SA-Configure database is opened locally. In prior versions, the L*i*SA-Configure database was opened with a server name set, resulting in network traffic and sometimes the need for connection document to the local machine.

In Cluster environments, the search sequence for the L*i*SA-Configure database was changed. The module uses the following sequence :

1. current directory on current server
2. Lotus Notes data directory on current server
3. Lotus Notes data directory on cluster mates

The new sequence results in a better performance if the L*i*SA-Configure database is not located in the current directory.

The alias was splitted into a design-alias specifying the template and a database-alias specifying the replica copies of the database. This behavior works only with a L*i* SA-Configure V2.0 database. If you work with an older design of the L*i*SA-Configure database, both new alias names are the same and behave like the old single alias name.

Two new functions added : DBAliasByPath and DesignAliasByPath.

If the L*i*SA-Configure database could be opened, but the needed view was not found, there was no error message.

- Version Control Tool (VCT) - Module :
  The VCT functions now support Notes Release 5 design elements.

- **Bug fixes**
  - SetClipboard :
    Corrected behavior if Rich-Text was copied to clipboard, then a text was copied to the clipboard and the pasting was performed in a Rich-Text field.
  - LSX - Module :
    A Memory leak was removed.
  - S/390 :
    A load bug from version 3.2 e was removed. Under certain circumstances the LSX module could not be loaded.
  - Sun Sparc :
    UDF crashed if return value was empty.

Things you need to know

CD structure

The CD contains various subdirectories as explained below.

L*i*SA 2.0

this release Notes

Windows

install routine

UDF

Win32

readme

Library

DLL files

Install

install routine

OS2

readme

Library

DLL files

AIX

readme

Library

DLL files

Tools

Load Test

SUN

readme

Library

DLL files

S390

readme

Library

DLL files

AS400

readme

Library

DLL files

NotesData

LSA

EN

L*i*SA-Templates LSA*.ntf

Demo

L*i*SA-Demo databases LSA*.nsf

DevPk

L*i*SA-Development build and library databases lsb*.nsf, lsl*.nsf

L*i*SAScripts

include files *.lss (mount as your S-Drive during development to access these as include)

(DocuPk)

L*i*SA-interface definitions as .nsf and .pdf

LSAScripts

HTML documentation for the above script libraries.

(the documentation is focussed on the most common routines

only)

(Homepage)

L*i*SA Homepage.nsf

(Pres)

various items usefull for presentations and live demos e.g. Parameter import files.

IDs

for your convenience the IDs used in the demo databases are also provided. The password for each user ID is "password".

This allows you to easily check out acess rights and experience the functionality. Also the L*i*SASave.ID for Cross certification is provided here.

NotesSmarticons

for L*i*SA-Help
for L*i*SA-DesignTracker
for a DesignTyper utility

Tools

L*i*SA-DesignTracker

readme
DesignTracker.exe

LSDoc

readme
install routine

L*i*SA makes use of other licenced material under the terms and conditions included in the Licence directory.

Installation

- For Windows an installation routine has been provided, that guides you through the installation process. Start the program `setup.exe` in the `Windows` subdirectory of the L*i*SA CD. The install routine for L*i*SA databases will install by default into the following directories
  - <NotesData>\LSA\EN\Demo
  - <NotesData>\LSA\EN\DevPk
  - <NotesData>\w32
  - <NotesExecutable>

- For other platforms all material has been provided in the Lotus Notes data directory as described with the content of the CD. For installation please copy the files onto your system. If you keep the same directory structure, the demo will be readily available. If you change the structure you may need to reconfigure the file documents in the L*i*SA-Configure database.
  HINT:
  Do not forget to reset the "read only"-flag in the files you copied from the CD!

- For the other server platforms, where you typically install the UDF-modules you will find the operating system versions in the operating system sub-directory within the UDF directory. A readme file gives further instructions.

- Cross certify your Domino server with the L*i*SA Development ID. Please read the details in the L*i* SA-Help documentation in the chapter "L*i*SA Developer Pack / Installation".

- There is a tools directory on the CD that provides you with the L*i*SA tools. The L*i*SA-DesignTracker helps to visualise the inheritance structure of the design of all databases that are available on the given servers, on the local disk or in customised project folders. Furthermore, it offers a set of functions that refresh database design properties. It is usually copied into your Lotus Notes and

Domino executable directory.

- The LSDoc tool generates HTML Documentation from exported LotusScript files. Please follow the instructions in the readme.

- To setup a web browser demo environment, a special homepage database is provided. To install this, please backup your `homepage.nsf` database and replace it by the one provided on the `CD:\notesdata\lsa\en\homepage` subdirectory.

Migration

This chapter explains how to migrate from LSA V1.1 to L*i*SA V2.0.
The updates are provided as a Notes-Templates. You usually would refresh or replace your database design. Cases have been reported, where the replace did not fully operate on all design elements. Therefore it is recommended to proceed as follows

1. Backup your customized subforms LSCxxx or any other changes that you did.
2. Replace the design by the following BLANKO.NTF by File->Database->Replace design.
3. Check that all design element have been deleted. If necessary delete the design elements manually.
4. Replace the new design by File->Database->Replace design.
5. Reintroduce your changes.

The version has been designed to be upward data compatible whenever possible. There are few places where additional capabilities required changes. In these cases migration agents are provided:

- **L*i*SA-Configure**

The field DesignAlias is new in L*i*SA 2.0. It must have the value of the old DbAlias field. The value of DbAlias is kept, so that DesignAlias and DbAlias contain the same values. The agent "LSAMigrationAgent 1.1 --> 2.0" changes L*i*SA 1.1 configuration data.

- **L*i*SA-Organization**

Due to various improvements and new features the existing data from V1.1 L*i*SA-Organization databases has to be migrated. The agent "Migration to 2.0" changes LSA V1.1 organization data.

- **L*i*SA-Footer**

Issues with footers in an international environment have been addressed. The Footer now adopts to the date settings of the user. The agent "LSA Migration Agent Footer 1.1 --> 2.0" analyzes the footer history and puts it into the new format. Given the textual date cant be interpreted, the data will be preserved. If the setting is to delete history data beyond a certain period, December 30, 1899 will be used.

- **L*i*SA-Address**

The parameter LSAAddrPhoneFormat uses new keywords as defined below. The old keywords (given in brackets) are discontinued but mapped accordingly for compatibility.

For the interface use the following keywords to build the format

CountryCode (PCountry)	->	country code of the number
AreaCode (PArea)	->	area code of the number
Number (PLine)	->	number
Extension (PExtension)	->	the extension of the number

Limits

- **Browser support**
  - L*i*SA has been tested with

- Microsoft Internet Explorer 5.00
- Netscape Communicator 4.73

and a variety of other Releases. There are minor differences in the representation of the L*i*SA Layout (colors, positions, etc.). The L*i*SA Development has made any efforts to work around these incompatibility. To mention just a few:

- Internet Explorer Version 4.72.2106.8 doesn't open a window using LiveConnect
- Internet Explorer Version 4.72.2106.8 doesn't repaint applets correctly
- Netscape does may display the applet background color grey instead of the correct white background.

- **Domino Limits**
  - LiSA is based on Domino R5, which has a documented limit of 15 KB for Fields. Although any precaution has been made in order to work arond this limit you may experience limits in the number of members, the parent units or traits you can use in the LiSA-Organization Unit. Likewise the number of levels in a LiSA-Glossary or the length of parameters in LiSA-Configure may expose these limits. Usually these Limits have litte relevance in practice.

Troubleshooting

- **L*i*SA-Organization**
  - recursions must be removed by manually removing the unit which caused the recursion as member from the higher level unit, starting with the lowest level unit which shows a recursion, always followed by a database update
  - lock document must be removed manually if any background agent failed, see view "Administration/LOCK Document" shows update request documents instead
  - LSAOrgUpdatesDone has max. 20 entries