



COMPUTER LINE ASSOCIATES S.r.l.

Via della Viggioletta, 8 - 29100 Piacenza
Tel.: 0523/497066 – fax: 0523/497713 – web site: www.cla-it.com

SPIDER

Management and control of data relating to project items

**White Paper
April 2009**



INDEX

SUMMARY	3
WHY USE SPIDER?	4
ADMINISTRATIVE TOOLS.....	5
SpiderTemplateBuilder	5
Insertion of a configuration comment	6
Loader	7
SPIDER FOR END USERS.....	8
OPERATION	9
General Data.....	9
Associations	10
Permissions	11
Structure of the Datasheets	11
PROJECTS	13
Cancellation of the item	14
Notification	15
To Do List	15
Revision	16
Comparison of datasheets	17
Principal functions within the Datasheets.....	18
Messages and notifications	19
CONCLUSIONS	20

Summary

'Spider' Engineering Data Warehouse is an application that arose from the need to introduce a simple and effective solution into companies aimed at the management and control of the integrity of data and of the workflows that contribute towards the complete definition of the characteristics of the Project Items.

The system ensures that the consistency of the data is kept constantly up to date, preventing their duplication and gathering them in a structured way in datasheets defined by the Engineering Company itself, furthermore allowing the rapid and efficacious exchange of information between the users involved in the Project.

The structure of the datasheets (configured and imported into 'Spider' through the use of the 'Spider Template Builder' tool) is parameterizable and adaptable to all types of situations; the application envisages the setting of some configurations pertaining to the propagation and management mode of the data which allow the company dynamics to be reflected.

The principal functions of 'Spider' are:

- personalized configuration of the application;
- definition of the base tables;
- association of the Project entities (contexts, templates, items);
- automatic management of the locking of the documents during editing;
- sharing of the information published in the Project datasheets (the system is able to manage the possibility of a proposed modification to the data by another user, by means of an automatically generated notification);
- automatic alignment of the Project datasheets through the data propagation flow set during the start-up phase of the Project itself;
- management of the "events" generated by the modification of certain fields of the datasheets defined as "sensitive", that is, which contain data whose variation could imply the adjustment of other data entered in the datasheets that have imported the item.
- comparison between datasheets in an individual Project, (current document/revision, revision/revision);
- exportation of the datasheets in Excel files with the possibility of sending them directly - by e-mail - to any address indicated by the user;
- maintenance of the history log of the datasheets subjected to one or more revisions.

Why use Spider?

The complexity and the difficulties encountered by engineering companies in the management of the exchange of information relating to the definition of Project items inevitably lead them to orient their strategy towards automated and structured management, which guarantees the total sharing and accuracy of the information at all times.

'Spider's' strong suit is its ability to guarantee the consistency of the data and of their flow, which, given that the data come from different sources, are potential sources of error.

'Spider' is able to:

- eliminate the criticality produced by the divulgation of hard-copy documents, a situation that generates confusion and misalignment;
- guarantee the quality, uniqueness and traceability of the data;
- optimize the time required for the transfer and completion of the Project datasheets;
- ensure the uniformity and the immediate updating of the information;
- introduce a type of communication open to all the personnel involved in the process;
- monitor, at all times, the status of the activities relating to the definition of the items destined for the construction of the plant.

By means of automated checks, the system detects all the modifications brought to bear on the Project datasheets and, following predefined logics that vary according to the type of datum, the operating mode and the type of propagation set, it manages their alignment.

The updating may be:

- **'Immediate'**: the variation of the 'item code' (which may be performed only on the parent document; that is, that in which it is coded) is propagated directly in all the datasheets (direct or indirect children) in which it is present.
- Performed upon the **'Revision'** of the document (envisaged within the 'From Revision' operating mode which sets the possibility of importing items solely from revised documents).
- Performed both upon the **'Publication'** and the **'Revision'** of the data (in the 'From Current' operating mode which accesses the importation of the data from the saved current documents).
- In **"pending"** status, that is, awaiting confirmation from the user who owns the data subjected to modification (when the modification of owner-data is foreseen).

For both operating modes, the propagation of the data may follow the flow:

- **'To All'**: acts on all the datasheets that have imported the item both directly and indirectly;
- **'To Child'**: acts only on the datasheets that have imported the item directly; those that have imported it indirectly will be updated only upon the publication/revision of the importation datasheet.

During operation, it is possible that the modification of some information may significantly influence the data entered subsequently; for this reason - in the initial analysis phase - the company may indicate certain cells to be defined as "sensitive" in that their variation entails the user checking the subsequent definitions applied to the underlying documents. This type of "event" triggers the automatic generation of the 'To Do List', which will contain the details of the modification and allow the users involved to evaluate its impact on their own project documents.

The company may also indicate cells where, following the integration or the variation of certain data related to the item, the system will have to eliminate the content in that it will be reformulated on the basis of the new indications.

Administrative tools

'Spider' resides within a *Suite* of software applications aimed at EPC companies, which allow the simplification of some processes linked with the design, realization and maintenance of industrial plants.

Within the *Suite*, the installation of a '*Database Manager*' administration Console is foreseen (aimed at IT Managers, key Users, etc.), which is dedicated to the management of the information common to all modules.

In essence, '*Database Manager*' constitutes the control panel that allows the company to concern itself from the beginning with the configuration, the management and the control of the base tables of the following entities:

- Base of Data
- Modules
- Users
- Groups

It is also dedicated to the:

- Creation of the Contexts (environments within which the projects are developed, for example, on the basis of the reference standards used: ISA, KKS, etc).
- Creation and maintenance of the Projects.
- Definition of the roles and of certain permissions to be assigned to the users.
- Importation and exportation of the data within a Database schedule.

The use of the Console allows administrative-type activities to be converged into a single tool, optimizing the management of the computer supports, guaranteeing the uniqueness of the data and minimizing the time required for general configuration.

SpiderTemplateBuilder

The '*SpiderTemplateBuilder*' tool is the instrument used to load the destination database with the Excel files that are used for the structuring of the templates.

The interfaces that contain the configuration data are organized in sections (every section coincides with a sheet):

- **TEMPLATE_INDEX.xls**: represents the complete archive of all the datasheets that must be loaded into the database schedule; for each section it is necessary to insert the pertaining datasheets.
- **TEMPLATE_SECTIONS.xls**: this file is used for the definition of some parameters to be applied to the sections of the datasheets.
- **TEMPLATE_COLORS.xls**: manages the colours of the fields of the datasheets. The use of colours is intended to allow easy orientation within the document as they allow the immediate identification of the types of cells displayed.
- **MAP_TABLES.xls**: contains the configurations of the service tables, for the purpose of populating the '*Spider*' application with the necessary user information for correct operation. Some tables may not be displayed, in that their contents can also be inserted directly from within the application; others, on the other hand, are indispensable for the use of the system.

Before proceeding with loading the datasheets within the desired schedule, it is necessary to carry out the configuration of the templates (following certain conventions).

In this phase, it is necessary to:

- have Excel installed on the work station you are operating on;
- have at least one 'Template_Color' defined;
- have the structure of at least one section of the datasheet completed;
- have defined the types of data that can be associated with the cell.

The procedure to follow for the compilation depends on the type of datum that the cell will contain; the types of datum envisaged are shown in the table below.

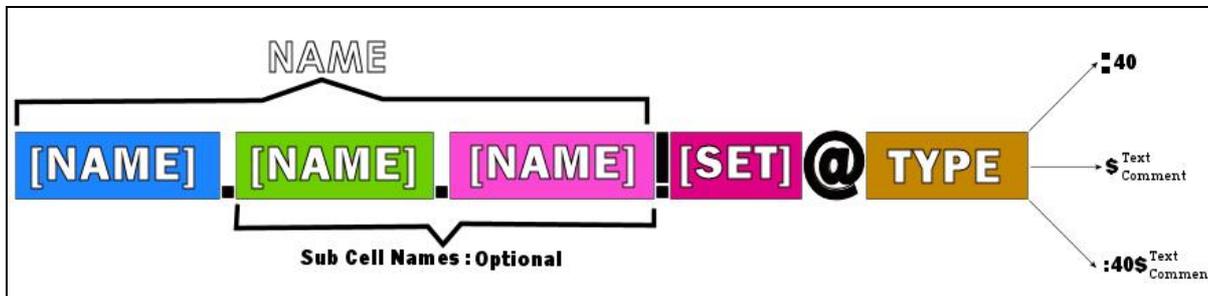
Type of cell	Description
INT	Whole number.
TEXT	Text box.
FLOAT	Number with floating decimal.
DATE	Date.
GLIST	Global List (pull-down menu).
PLIST	Project List (pull-down menu).
ITEM	Item.
PLINE	Project Line.
PROJECT	Datum of the Project (e.g.: project code).
DOCUMENT	Datum of the document (e.g.: header data).
UOM	Unit of measurement.

During the configuration, it is necessary to attribute a name to the cells (UDA - User defined attribute) which it is intended to relate to each other, this name must observe a special formatting whose first part is the same for all types of cells and which differs in the end part according to the type to which it belongs.

The name of the cell and its morphology are configured within the cell comment.

Insertion of a configuration comment

The diagram below illustrates the structure envisaged for the correct configuration of the cells within the datasheet:



Example:

[AMBIENT_CONDITIONS].[OTHER_VALUE].[#]@TEXT:80

[AMBIENT_CONDITIONS] à Name

[OTHER_VALUE] à SubName

[#] à Set

TEXT à Type

80 à Length

. ! @ : à Separator characters

The data envisaged concern:

Value	Description
Name	Attribution of a Name to the cell; this must be contained between parentheses [].
SubName	Optional value; in any case, one or more SubNames may be inserted according to the utility; each SubName must be contained between parentheses [] and separated by a full stop.
Set	Obligatory indication; indicates the group of cells to which the cell belongs and must be the same for all the cells that it is intended to manage within the same group. Also in this case the value is inserted between parentheses [].
Type	Obligatory value; this must be envisaged among those provided by the system (see types in the data-type Table shown above). If a type not envisaged among those indicated is required, it is necessary to contact the system administrator.
Length	Optional value; useful if it is wished to specify the length of the field. It is applicable solely and exclusively to “text”-type cells.

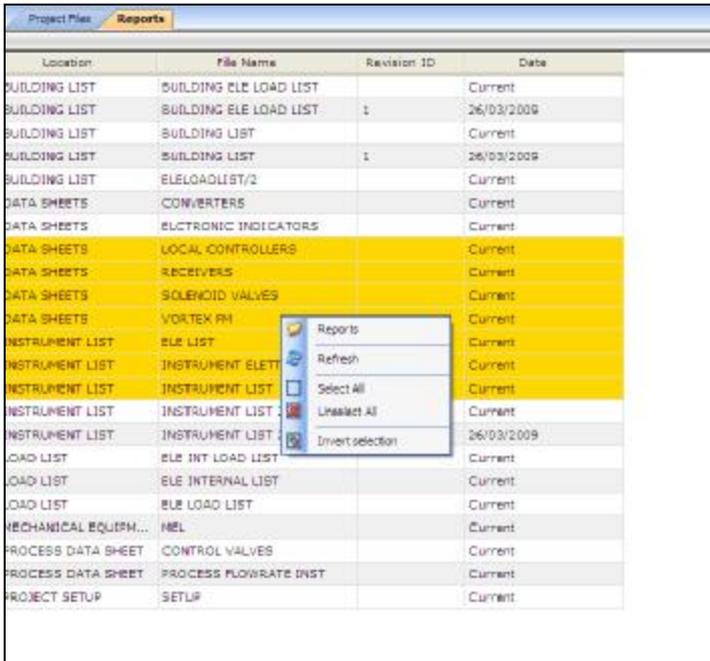
Loader

The function dedicated to loading the templates into the ‘Spider’ database manages the various operations using a wizard-style interface; this allows a guided path to be followed to load the configuration files.



Spider for end users

'Spider' uses the standard Windows interface, which is now common to all company PCs, guaranteeing users simplicity and immediacy in its use. Being integrated with MS Office, the documentation is easily exported in Excel format.

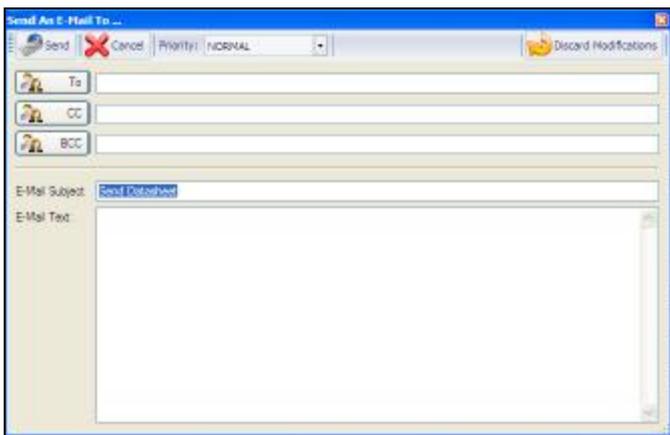


The screenshot shows a window titled 'Project Files' with a 'Reports' tab. It contains a table with the following columns: Location, File Name, Revision ID, and Date. A context menu is open over the table, showing options: Reports, Refresh, Select All, Unselect All, and Invert selection.

Location	File Name	Revision ID	Date
BUILDING LIST	BUILDING ELE LOAD LIST		Current
BUILDING LIST	BUILDING ELE LOAD LIST	1	26/03/2009
BUILDING LIST	BUILDING LIST		Current
BUILDING LIST	BUILDING LIST	1	26/03/2009
BUILDING LIST	ELELOADLIST/2		Current
DATA SHEETS	CONVERTERS		Current
DATA SHEETS	ELECTRONIC INDICATORS		Current
DATA SHEETS	LOCAL CONTROLLERS		Current
DATA SHEETS	RECEIVERS		Current
DATA SHEETS	SOLENOID VALVES		Current
DATA SHEETS	VORTEX PH		Current
INSTRUMENT LIST	ELE LIST		Current
INSTRUMENT LIST	INSTRUMENT ELET		Current
INSTRUMENT LIST	INSTRUMENT LIST		Current
INSTRUMENT LIST	INSTRUMENT LIST		Current
INSTRUMENT LIST	INSTRUMENT LIST		26/03/2009
LOAD LIST	ELE INT LOAD LIST		Current
LOAD LIST	ELE INTERNAL LIST		Current
LOAD LIST	ELE LOAD LIST		Current
MECHANICAL EQUIPH...	MEL		Current
PROCESS DATA SHEET	CONTROL VALVES		Current
PROCESS DATA SHEET	PROCESS FLOWRATE INST		Current
PROJECT SETUP	SETUP		Current

Direct exportation in Excel file

Integration with the company's messaging system and a notification system within Spider mean that internal users of the process can be informed promptly on any variations made to the data.



Direct sending of e-mails

Operation

Subsequent to the installation of *'Spider'*, after configuring the Excel files from which the Project datasheets are extrapolated and after the administrative activities to be performed on the *'DbMan'* console, it is possible to access *'Spider'* and proceed by following the logic path illustrated below.

General Data

In this section, the base tables for the definition of the entities used within the Projects are defined.

- **'Type of Data'**: units of measurement used in the datasheets, correct configuration is essential for the functioning of the application in that the system is based on the information supplied for internal conversions.
- **'Global List'**: represents all those items that make up the menus of choices available in some fields of the datasheets for which you can set the possible associated selections in advance.
- **'Unit of Representation'**: definition of the units of representation to be applied to the rapid choice menus present within the datasheets, such as: language and unit of measurement.
- **'Item Types'**: definition of the types of items used within the system.

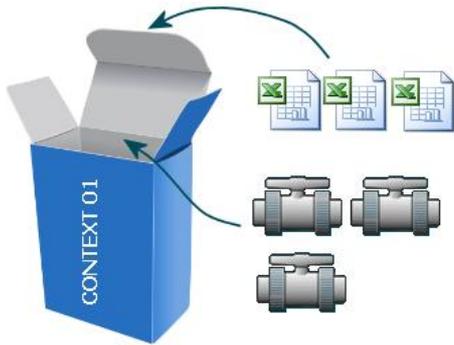
UOM Type ID	UOM Type Code	UOM Type Desc
2	AREA	AREA
13	CAPIRE_LATERIE	CAPIRE_LATERIE
16	CONDUCTIVITY	CONDUCTIVITY
19	CURRENCY	CURRENCY
6	DEBITA	DEBITA
14	EL_POWER	EL_POWER
7	LUMINCO	LUMINCO
15	MOMENTO	MOMENTO
18	PCSC	PCSC
7	PCSC_MOLECOLARE	PCSC_MOLECOLARE
3	PCSC_SPECFIO	PCSC_SPECFIO
16	PONATA	PONATA
20	PRESSONE	PRESSONE
21	PREVALENZA	PREVALENZA
11	RUMORCOSTA	RUMORCOSTA
17	SPED	SPED
12	TEMPERATURA	TEMPERATURA
9	TENSORE_GESNOLI	TENSORE_GESNOLI
9	VEL_CARTA	VEL_CARTA
4	VISCOSITA	VISCOSITA
5	VOLUME	VOLUME

UOM ID	UOM Type ID	UOM Code	UOM Desc	UOM Symbol	UOM Mul Entry	UOM Add Entry	UOM Partial Rate
10	1	VC1	CM	CM	0.1	0	2
17	1	VC2	CM	CM	0.00001	0	2
18	1	VC3	M	M	0.001	0	2
19	1	VC4	MICRON	MICRON	1000	0	2
20	1	VC5	Mil	Mil	1	0	0
21	1	VC6	INDI	INDI	0.00001	0	2

Insertion and configuration of the units of measurement

Associations

In the system, every Project must of necessity be associated with a reference Context, within which the links between the various entities will be defined. The Context represents the environment where the Project itself is developed, for example, on the basis of the reference standards applied (ISA, KKS, etc.).



Association (1)

Firstly, it is necessary to associate the Items present in the system and the Templates to the relative Context, this way all the Projects that have a certain Context associated will automatically inherit all the information present in it.

The Context - once it has been created and all the associations are completed - may be used for all the Projects the user requires; creating well-organized Contexts allows the preparation phase of a Project to be speeded up.

For example: if the user avails of Templates created in different languages, he or she can create specific Contexts for each language, or configure Contexts that group underlying data with specific standards, etc.

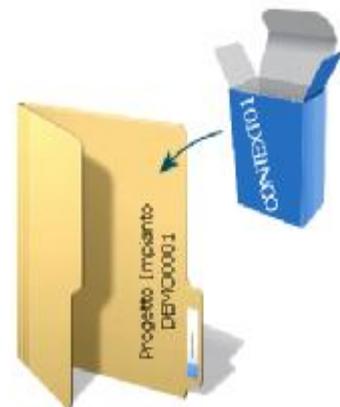
Association (2)

Subsequently, it is necessary for all the Items and Templates present in the context to be associated, that is, every Template must indicate which Item types may be dealt with inside it.



Association (3)

Having completed all the associations and having configured the Contexts, it is possible to proceed with the creation of the Projects, which will automatically inherit their settings.



Permissions

The attribution of the permissions relating to the operation of the *'Spider'* users is managed through the DBMan administration console, and is based on the concept of permissions assigned to users and groups according to their roles.

The management of the roles is utilized to allow the differentiation of the permissions by which the forms and functions of the application can be accessed; by means of this management it is also possible to diversify the accesses according to the base of data and/or the project to be used.

The fundamental concept governing the assignment of the permissions is that roles are attributed to every user or group; each role is subsequently associated with one or more permissions.

Structure of the Datasheets

The datasheets are laid out in a double structure: the first part dedicated to the creation of the item, and the second to the importation of the item previously coded by another user, allowing the addition of the user's own data.

Those most commonly used can be divided into two types: "grid" and "flat".

"Grid type": is composed of a number of sections defined in the project (initial configuration), it is not possible to add others; in the event that this is necessary, a new template must be formulated. The data are inserted in a grid and populate from the top to the bottom, each row may contain only one item, which can be:

- Coded: in the event that the item is being generated
- Imported: in the event of inheriting the item previously imported by another user.

"Flat type": contains from 1 to "n" items; when the maximum limit of data inserted is reached, it is possible to add a new section, always duplicating only the last one available. The structure is contained in an A4 sheet, in this format the data are inserted in scattered mode, and in this case, too, the item may be:

- Coded
- Imported

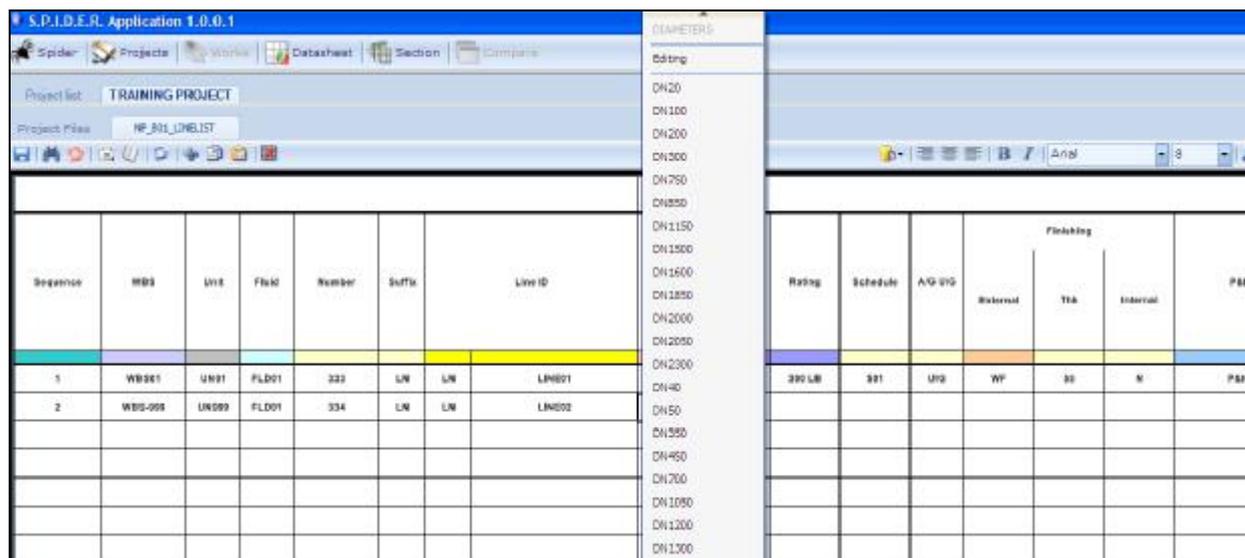
For both types, the fields present may be:

- **"writable"**: fields of the document in which the data are inserted manually;
- **"manual import"** fields: typical of the cells that contain items generated in other datasheets, the item is chosen, imported in its own document and can never be modified (it is only possible to eliminate it from its own document by using the "Detach this Item" function).
- **"automatic import"** fields: the datum is inherited from the importation of the item that was carried out previously. This type of import may take place in various stages; that is, the fields valorised in the source datasheet and common in both documents are immediately populated, whereas some definition fields of the same item which depend on underlying datasheets will be updated when they enter and form part of the flow (that is, they are compiled and published/revised).
- **"choice"** fields: by means of pre-set menus (Global Lists); in this case, the data reside in certain tables populated during the configuration phase of the application. Should the value concerned be absent, it is possible to gain access directly to its insertion by using the 'Editing' item present on top of the data list displayed).

The principal data of the Project (Project name and number, Client name and number, location, etc...), which are present in the header of every datasheet, are populated upon the compilation

and issue of a template dedicated to the setup of the Project itself (typically A01-PrjSetup); any integrations or modifications made over time to this document (non-duplicable) will automatically be propagated in all the datasheets.

Within the templates, some colours have been used allowing the user to orient himself or herself more easily inside the document, and which allow the immediate identification of the types of cells displayed.



Layout of grid type datasheet (with GlobalList opening)

Generally, the colours are:

- bright yellow: identifies the cells where the item is coded;
- bright pink: applied to cells defined as "sensitive"; that is, they trigger "events" in the 'To Do list';
- pale yellow: indicates the cells owned exclusively by the datasheet; the content of these fields is created and remains solely within the document (it is never imported);

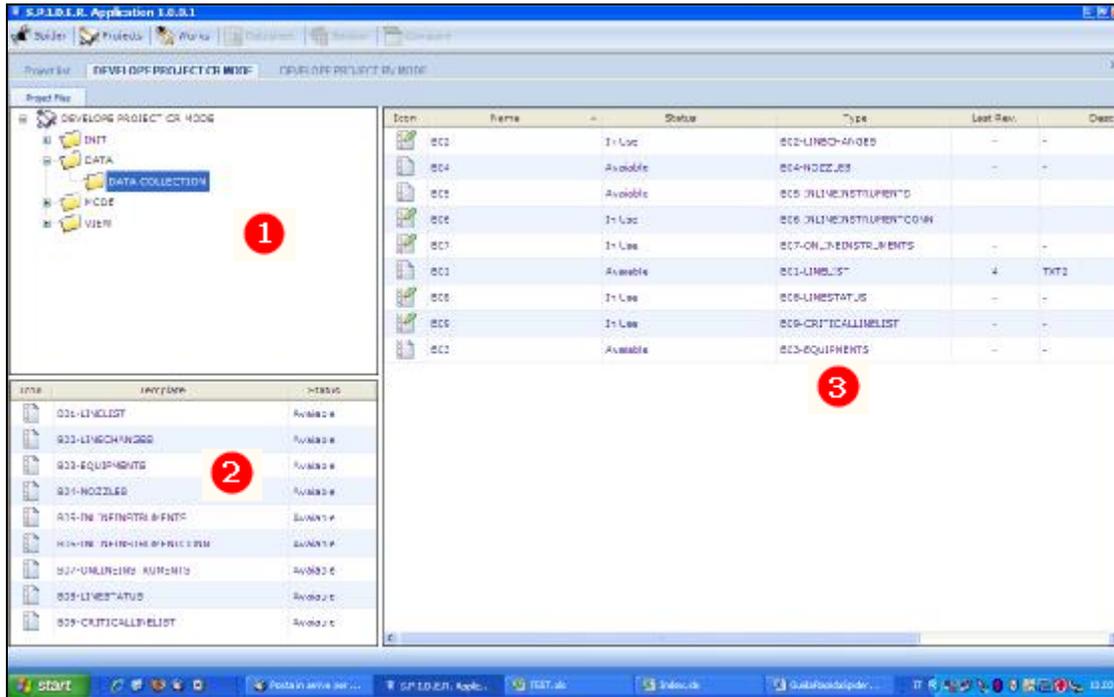
The other colours used to identify the type of fields available in the datasheets are defined on the basis of the requirements of the company (for example, cells for importation of the item itself, cells for the importation of specification data of the item, cells for the insertion of new details for the item imported, etc.).

		JOB	SP Number	Client Number
		TR01		
<u>Fluids</u>				
		Fluid	Description	
1	FLD	FLD01	FLUID 001	
2				
3				
4				
5				
6				
7				

Layout of flat type datasheet

Projects

This section is the heart of the application, where the definition of the items is developed through the compilation, the sharing and the propagation of the information of the Project datasheets.



Display of the Project

The display window of the Project is divided into three sections: the first (1) contains the tree-structure of the project and represents the organization of the datasheets inside the system; the other two are empty and will be compiled by expanding and selecting the underlying nodes.

Section (2) is a list of the templates available for the creation of a new datasheet, with an indication of its type and status:

Status	Description
	available for the authenticated user.
	locked for the authenticated user.
	available in read-only mode for the authenticated user.
	template for which the authenticated user is enabled to receive notification e-mails relating to the datasheet itself.

Section (3) contains the list of the datasheets belonging to the type of the folder selected in the frame (1); in this case the use of the datasheet is subsequent to its creation, hence it will be accessed to integrate data or modify them. In this section the datasheets, as well as the statuses already displayed, may assume the following additional values:

Status	Description
	in use by me
	in use by another user

In order to be able to operate on the datasheets, it is necessary that they are available, that is, they are not in use by another user (in any case, the datasheet will open in read-only mode).

Having defined the structure of the various templates, having performed the various associations, created the project, assigned the permissions, defined the operation (From Revision or From Current) and the data propagation mode (To All or To Child), the datasheets are made available to the users of the system who - with a simple operation - may encode the items or import them into their own datasheet to specify the parameters better.

In the presence of datasheets that import items, the system accesses the list of the documents that contain compatible items and displays only those that are still available, this means that they will have been filtered using association with the templates and hiding the items imported previously. Within the system the item is available until the user saves the document; upon its first saving it becomes his property and thus is not available for other users at the moment. In the event that another user had imported the item, the first person to save it acquires its ownership, the other - upon saving - receives an error message (hence it is advisable to save the datasheet immediately after importing data). This check helps avoid circumstances whereby two users work simultaneously on the same item.

Depending on the type of operation chosen, once the datasheet has been saved or issued (in the latter case, by means of the revision function), the item and all its accompanying data become part of the system; at this stage all the information may be shared among all the users involved in the Project. They may recover the data by transferring them into their own datasheets, complete them with other information and make them available once again for subsequent integrations.

The imported item drags with it some cells (information) that are entered in the source document; those that coincide (source cell = destination cell) will be compiled automatically, and will be updated every time a revision of the source datasheet is issued, in the 'From Current' mode also upon its publication.

Cancellation of the item

The system allows the definitive elimination of a created item only in the event that:

- it has never been revised;
- it has not been imported into other datasheets.

Cancellation eliminates it definitively from the database, making the related code available once again.

In the event that the item has been previously revised, but not yet imported by other users, the cancelled record is displayed in crossed-out grey text, this indicates that the item can not be used, thus it is not enabled for typing and can not be imported into other documents (the item code results as "not available" in that it is not eliminated from the database).

If it becomes necessary to use the cancelled item once again, the system allows it to be restored through the use of the 'Resume' command.

Notification

The "Notification" is an automatically generated message sent via e-mail to the enabled users of the system. This notice informs the user on what is happening to the document; for example: if within the current datasheet data have been updated that relate to the imported item and this item has been exported in other underlying files, the notification will advise of the necessity to issue a revision or produce a publication to propagate the new information.

The notification may also have been generated by the modification of one or more fields belonging to the source datasheet, by the user who imported the item. The document containing the modified item enters into a "pending" status, preventing it from being revised or published until the owner of the datum, by means of a specific procedure, accepts or refuses the proposal. In some company realities the possibility of modifying data belonging to another user is not foreseen (the imported cells are in read-only mode), in this case this type of notification is absent.

To Do List

The 'To Do List' is the function that manages the modification of some fields of the datasheets defined as "sensitive"; that is, which contain data whose modification could imply the variation of other data entered in the datasheets that have imported the item to which it is associated.

The section presented lists, in a table, all the modifications made over time on the "sensitive" fields of the project datasheets, reporting the following information:

- 'Date': date of modification;
- 'Item Code': code of the item on which the modification has been made;
- 'From User': user that made the modification;
- 'From meta doc': name indicating the type of template;
- 'From doc': name of the project datasheet on which the variation has been made;
- 'From section': name of the section in which the modification has been made;
- 'From Cell': name of the cell involved in the modification;
- 'To Document': destination datasheet;
- 'To Section': section of the destination document involved in the modification;
- 'To User': user name of the recipient of the notification of modification, who must verify its impact and subsequently assign a status to it;
- 'Status': status of the to do;
- 'Status date': date when the status was applied to the to do;
- 'Status Note': any applicable notes.

It is possible to change the status of the flagging and insert any necessary notes, directly open the datasheet that has received the modification, apply display filters and export the grid in an Excel file.



Change of Status panel of the 'To Do'

The 'To Do List' is also present in the individual destination datasheet, within which are listed only the 'to dos' relating to the datasheet selected.

Upon the occurrence of the "events"; that is, of situations that generate a 'To do', the system envisages the possibility of automatically cancelling some information contained in certain fields and defined on the basis of the imported data (for example, some data fields). Once the 'To Do List' has been compiled and the item that has been imported into the destination documents is intercepted, the application – following the logic flow defined beforehand by the company – is able to cancel the cells that should be reconsidered on the basis of the modified datum inherited from the import document.

Revision

"Revision" is the action that permits the issuing of the document with integrations or modifications carried out and hence its making available within the system; any variation is automatically applied in all the datasheets containing the information relating to the item subject to the revision itself.

To issue the revision, it is necessary to associate it with a "Revision Index", this allows the grouping of the datasheets which, for usefulness' sake, it is desired to issue within a single revision applying an identifying label to it within the system.

This same datasheet may be attributed to the same index once only, but may in any case be present in more than one index (that is, it may have several revisions associated with different indexes).

To issue a revision, the following conditions must be observed (verified by the application before proceeding with its execution):

- the document must be available (that is, not in use, neither by me nor by other users);
- the status of all the documents reporting the imported items must be "Available" (not in use);
- the datasheet must not have a "pending" status (situations of proposed modifications to one or more items of pertaining data not yet resolved)



Selection of the datasheets to be revised

The icons that represent the status of the datasheets indicate:

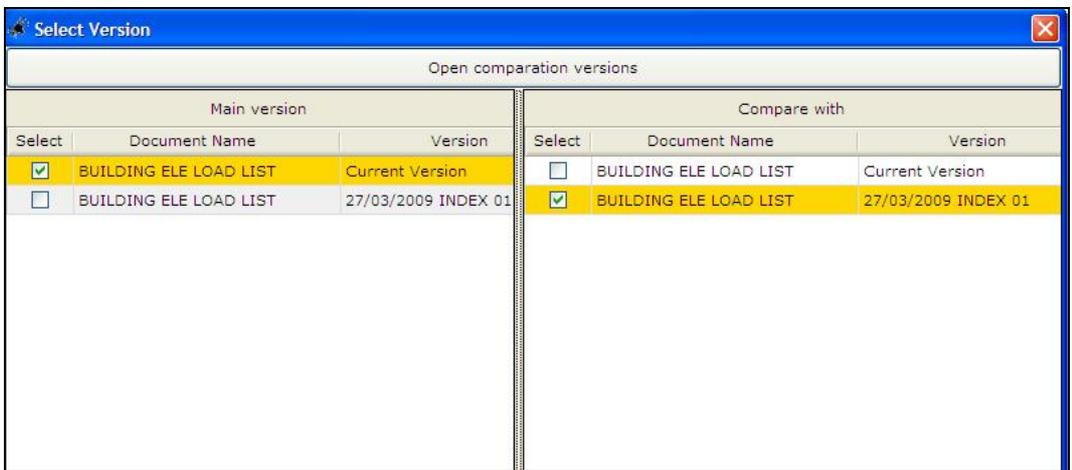
Status	Description
	Available, thus potentially revisable
	In use by me; to revise it, it must be released

	In use by another user; to revise it, the user who is working on it must release it.
	Datasheet for which the user is enabled for notification (when envisaged), potentially revisable.

The application automatically verifies if there are impediments to the issuing of the revision, in the event that it is not possible to proceed with its issuing, the causes will be indicated in the record of the locked datasheet.

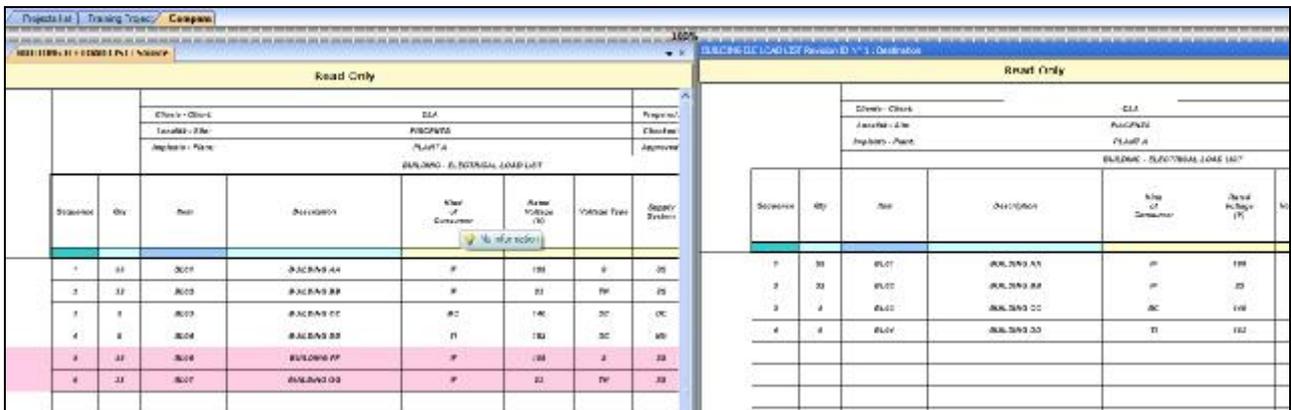
Comparison of datasheets

Within a Project it is possible to make a comparison between different versions of a datasheet, both between current and previous ones, and between two previous versions; the function is available only if there is at least one revision of the datasheet.



Vertical comparison display

The differences between the two versions are highlighted only in the reference datasheet (the one loaded on the left in the vertical display, or above in the horizontal display).



The screenshot shows two side-by-side windows displaying the 'BUILDING ELE LOAD LIST' datasheet. The left window is the 'Main version' (Current Version) and the right window is the 'Compare with' version (27/03/2009 INDEX 01). The left window has a 'Read Only' status and shows a table with columns: 'Scope', 'Qty', 'Rev', 'Description', 'Area of Consumer', 'Area (m²)', 'Voltage (kV)', and 'Supply System'. The right window also has a 'Read Only' status and shows the same table. Differences between the two versions are highlighted in the left window.

Scope	Qty	Rev	Description	Area of Consumer	Area (m²)	Voltage (kV)	Supply System
1	33	3001	BUILDING 04	IF	188	0	25
2	33	3003	BUILDING 08	IF	23	30	25
3	8	3003	BUILDING 02	IC	146	30	25
4	8	3004	BUILDING 08	IF	183	30	25
5	33	3006	BUILDING 09	IF	188	0	25
6	33	3007	BUILDING 03	IF	23	30	25

Display of the differences detected

Furthermore, the comparison may be exported in an Excel file and sent as an attachment to any e-mail address.

Principal functions within the Datasheets

As has already been stated, in order to be able to operate on a Project datasheet, it is necessary:

- that the user is in possession of the necessary permissions;
- that the datasheet is available; that is, it is not currently in use by another user.

The checks applied during the compilation of the datasheet concern:

- the uniqueness of the item within the datasheet;
- the compatibility of the data entered within the typology of the cell.

The exportation of the datasheets in Excel files, automatically saved in the folder indicated by the system administrator during the configuration/installation phase of the application, envisages certain necessary conditions:

- the user must have permission to write in the folder;
- the user must have writing privileges in the datasheet;
- the document must have been saved previously.

In addition to the standard functions present in any Excel file, such as 'Copy', 'Paste', 'Find', 'Insert row', etc., the datasheets also contain some commands that allow the management of the document itself within the flow. The most important are indicated below:

'Rename Item': this function is available only inside the datasheets that create an item, allowing it to be renamed by varying its code (never its Type Code). This operation should be performed with the utmost care as the modification is propagated in all the datasheets in which it is present, losing the source datum and also modifying the history log (thus also updating previous revisions).

'Show Cell Notify: (function used to resolve "pending" situations) details, if present, the variations made by other users on their data; for each proposed modification, the following information is shown:

Type of cell	Description
Proposed value	New datum proposed
From User	Name of the user who proposed the modification
On Item	Item subjected to the modification
UDA Name	System indication which identifies the cell where the modification was proposed
At Date	Date of insertion of the proposed modification

The user who owns the data must, by means of a simple procedure, refuse or accept the proposals displayed, while the system will automatically a message to the user who has proposed the variation of the data, notifying him on the status of the proposal (accepted, refused or awaiting further details).

The acceptance of the modification implies the revision or the publication of the datasheet to align the documents that contain the item in question.

'Cell Audit' and 'Datasheet Audit': the first command highlights all the changes that have occurred in the individual cell selected; the second the history log of the changes that have occurred within the entire datasheet.

Messages and notifications

Two levels of messages are used in Spider: the first makes use of standard e-mail managed through the company mail server, the second is an internal messaging system which makes it possible to communicate with all Spider users directly within the application.

The internal notifications within Spider permit the display of messages relating to some variations made to the project datasheets, they are configured in the DBMan administrative console.

The notification relating to the presence of messages is always displayed on the status bar, in the absence of new messages, the display reads:  You have 0 new messages, whereas if new notifications are present, the message changes to:  You have 3 new messages highlighted in red.

Conclusions

In synthesis, '*Spider*' allows the company to implement a process that is able to provide useful methodologies for the organization of the activities connected with the definition of the Project items, reaching great operating efficacy. The checking and divulgation of the data are completely delegated to the system; this allows notable savings in terms of time and expense by guaranteeing accurate planning.

The system proposed is absolutely flexible and thus easily adaptable to company strategies, using simple and intuitive procedures that introduce a protocol oriented towards the best possible organization of the various company functions.



Computer Line Associates considers that the information contained in this page is correct at the time of going to press. Said information is subject to change without notice and is derived from the descriptions of the technical products used. Computer Line Associates shall not be held responsible for any error contained in this document. Unauthorised distribution and transmission of this document by any means, photostatic or electronic or other, is prohibited. Microsoft and Windows are registered trademarks of the Microsoft Corporation. Other products cited are the trademarks of their respective owners.