

IGSS Definition – Multiple contributors to one project

Guideline for how multiple system designers can work seamlessly in parallel on the same IGSS SCADA project



Version: 1.0

Date: May 17, 2018

Contents

1	Introduction	2
2	Building a new project from scratch	3
2.1	Planning the project – Before you start designing (All).....	3
2.2	Prepare the base project in IGSS (Project owner)	4
2.3	Distribute (Project owner)	5
2.4	Work on subtasks (Multiple contributors).....	5
2.5	Export the subtask project files (Multiple contributors)	5
2.6	Merge all files into base project (Project owner).....	8
2.7	Post production (1 person)	9
3	Adding new content to an existing project (Multiple designers)	9
4	Project locking for adding new content (Multiple designers)	10
5.	Modifying existing content in a project (1 person)	11
6.	More information.....	11

1 Introduction

When designing larger IGSS SCADA systems in the Definition module it is possible for more than one system designer to work on the same project at the same time for higher efficiency.

A project with multiple simultaneous contributors does, however, require a structured approach as well as some precautions.

This paper provides recommendations and a step-by-step guide for the best way to plan and execute an IGSS project with several contributors.

Any information provided in this document is subject to change and is provided for information purposes only, and is supplied without liability for errors or omissions.

Due to the changing nature of the information contained in this document Schneider Electric shall have no liability whatsoever for any damages or losses arising directly or indirectly as a result of any of the information in this document not being accurate, complete or up to date.

You may face three different scenarios:

- Building a new project from scratch
- Adding new content to an existing project
- Modifying existing content in a project

2 Building a new project from scratch

With multiple contributors to an IGSS project, some tasks can be done by one system designer only, typically the project owner, whereas other tasks can be distributed to several others.

The following work flow and roles are suggested:

- | | |
|------------------------|--|
| 1. Project planning | All |
| 2. Create base project | Project owner |
| 3. Distribute | Project owner |
| 4. Work on subtasks | Multiple contributors |
| 5. Export subtasks | Multiple contributors |
| 6. Merge | Project owner |
| 7. Post production | 1 person only (project owner or contributor) |

2.1 Planning the project – Before you start designing (All)

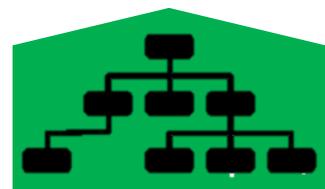
To ensure consistency throughout your IGSS project - and a good user experience for the final user of the SCADA system – the team needs to decide a common way to do things such as:

- **Naming and color convention** for objects, fonts, images, other files etc. Object naming: If several contributors need to add objects to the same areas, you should consider a naming convention to avoid conflicts. Conflicts will automatically be resolved in the merge phase but may introduce more work and/or bad object naming. See the ISA 101 standard for HMI for inspiration on use of color, navigation etc.
- **Alarm numbers and texts:** It is recommended that you pre-allocate the alarm numbers you plan to use, or alternatively allocate an interval of alarm numbers for each contributor to use. If not, any conflicts will be resolved in the merge by random reallocation of alarm numbers, and any structure you may have planned in your numbering will be lost. Alarm numbers to be used by many can be entered in the IGSS base project up front. See the ISA 18.2 standard for inspiration re. alarm management.
- **Alarm priorities and colors** should be also agreed upon before engaging in the system design to ensure that alarms are presented with the right effects and in a consistent way to the end user.
- **Drawing style** e.g. 2D (recommended in the ISA 101 standard) or 3D
- **PLC addressing rules**

See also [“Check list for planning a new IGSS project”](#) for more info.

Other preparations

1. Make list with object addresses and an I/O list for IGSS objects and atoms (tags)
2. Make a list or a simple “site map” of the SCADA project elements with main/overview diagram, sub-diagrams, graphs, dashboard etc
3. Agree who does what – and in which order. It is recommended to create all diagrams from the “bottom” and up.
4. Identify critical project parts and risks when more people start working on the project. E.g. is multiple languages needed?



2.2 Prepare the base project in IGSS (Project owner)

If multiple system designers are going to work on the same project at the same time, they can only work in the same *areas* of the project if you have prepared a few things.



System Configuration

All system configuration tasks can only be carried out by one user. In this phase you should create your server and eventual distributed driver stations with all drivers and nodes. This will make it possible for all contributors to add PLC addresses to their objects without entering the *IGSS System Configuration* module.

Symbols

Create all the symbols you may need throughout the project in *IGSS Definition*.

Individual contributors may add symbols when they do their subtasks. However, in case of conflicts when subtasks are merged into the project, conflicting symbols will be placed randomly in the symbol tables and you will lose control over the structure of your symbol tables.

You may, however, add symbols to the symbol file (.V20) and redistribute it to all contributors as long as you don't delete or change any existing symbols.

Animated symbols

Individual contributors cannot add custom designed animated symbols as these will currently not be included in the subsequent merge of all project elements.

As project owner you may, however, add new animated symbols and distribute the new animated symbol file (.CAT) to all contributors.

Scaling objects

You may prepare all scaling objects needed in order to reduce the risk of duplicates. Place the scaling objects in the area "Global" so they can be accessed by all contributors.

Protect objects

Create all protect objects needed so protection can be added initially by all contributors. Each contributor may add extra protect objects which will be merged into the project as any other object.

Templates for objects

If several contributors are going to share the same type of components (objects) you may benefit from creating templates for all these components in this preparation phase and place them in the area "Global".

If you configure all alarm numbers, scaling and protection on these templates, you will also reduce the need for the contributors to add these things to the objects.

If contributors are allocated complete areas they can also make their own templates.

Faceplates

Same considerations apply as for templates.

Multi-screen setup

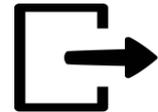
Use of multiple screens may be decided at this point of the project so that all diagrams can be placed on the correct screens by the contributors.

Image files

Consider allocating a sub-folder for each contributor in the IGSS project's image folder to avoid conflicts in image naming, which are tedious to resolve

2.3 Distribute (Project owner)

You can now distribute a copy of the base project's project folder to all contributors so they can build their part of the project. Before distribution, consider which tasks you allocate. You will reduce the work spent in the merge phase if you can have each contributor work in their own area.



NB! The smallest project fragment you can distribute as a subtask is a diagram. Several contributors cannot work in the same diagrams at the same time.

NB! When allocating subtasks, you should avoid that diagrams and areas of subtasks have cross references to areas and diagrams in other subtasks. References should be made after the merge.

2.4 Work on subtasks (Multiple contributors)

Save the base project files distributed by the project owner on your local work station.

Open the base project in the *IGSS Master's Design and Setup* tab, and carry on with your subtask in *IGSS Definition* – with due respect to the naming and color rules, alarm numbers, location of image files and other things agreed in advance.



Tip! When you create areas, diagrams and graphs as part of your subtask, there is no need to select "Name to menu" and/or "Description to menu" as this would have to be redone anyway in the merged populated project.

New Diagram Properties

Definition of New Diagram | Function Key Assignment

Name: Subtask Diagram B | @Subtask Area 1

Description: [Empty]

Window Properties

Caption: Name | Description

Title bar | Name to menu

System menu | Description to menu

Minimize | Visible in Mobile

Maximize | Keep window

Pinable | Status bar

Initially pinned | Popup

Keep aspect ratio | Fixed size

Diagram Type

Standard diagram

Faceplate master

Reusable diagram

Control group: [Dropdown]

Replicate group

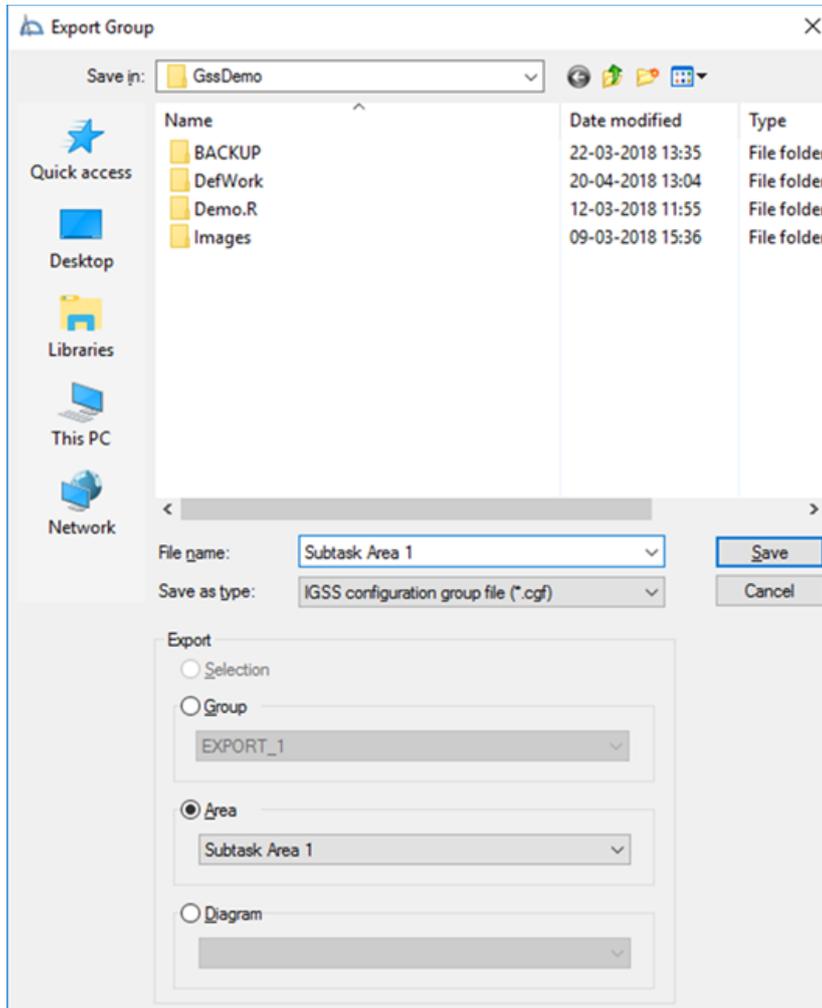
Background

Background Preview

2.5 Export the subtask project files (Multiple contributors)

After completion of the allocated subtask, each contributor should export the work. There are different ways to do this.

- If the task is isolated to an area, the contributor should simply use **File > Export** and select **Area** to generate an export file (.cgf), bearing in mind eventual agreed file and area naming.

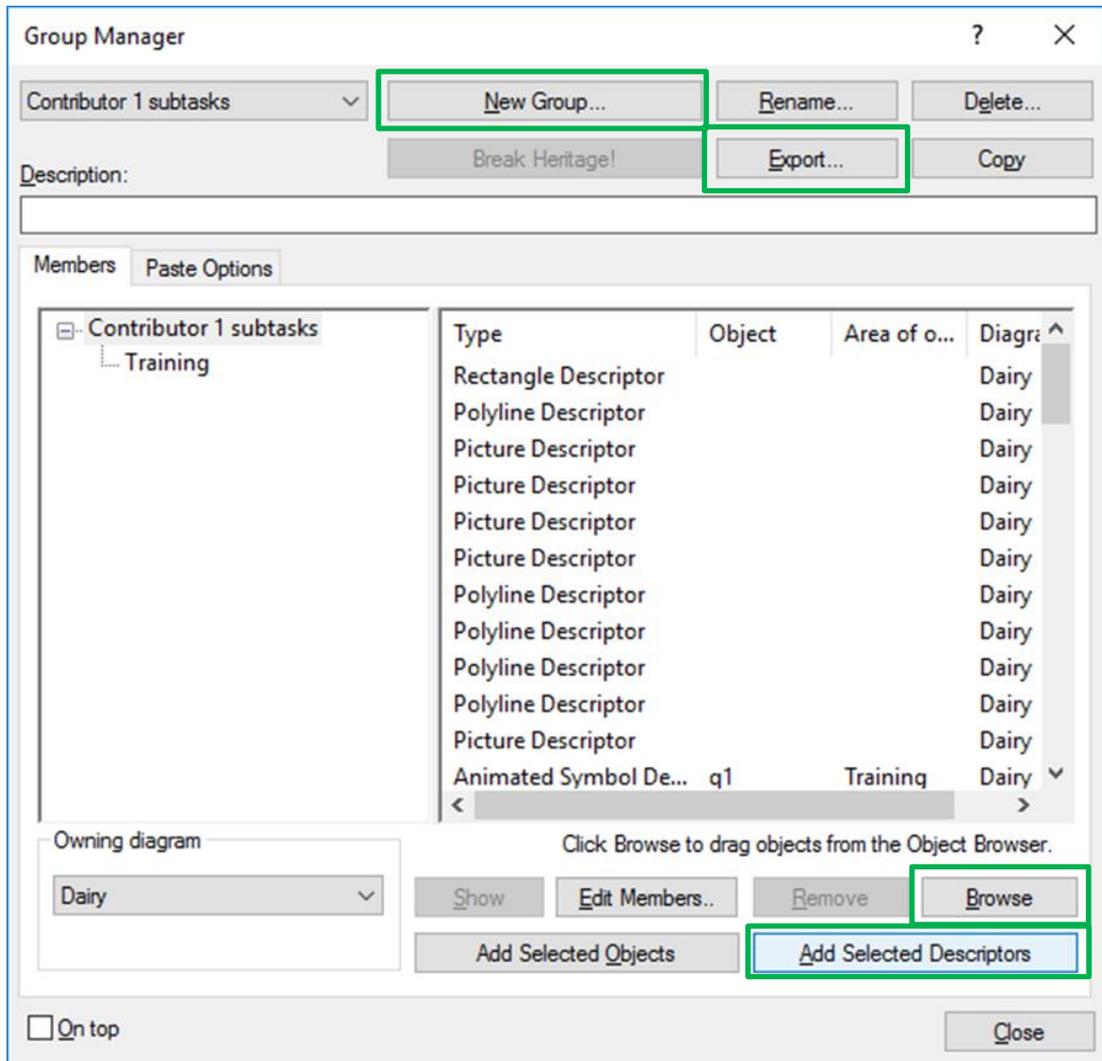


- If your subtask concerns one or more diagrams, you can use **File > Export** and select **Diagram**, which will generate one export file (.cgf) for each diagram.

NB! This will only work if you have NO references to other diagrams. If you have more diagrams with references between these, you must use the *Definition* module's **Group manager** for export:

1. In the *Definition* menu, select **Edit > Group Manager**
2. Create a new group in the Group Manager (see screen dump next page)
3. In the *Definition* menu, select and open the diagrams that belong to your subtask one by one, and repeat below process for each diagram:
 - Select all descriptors (CTRL-A)
 - In the Group Manager dialogue, press "Add selected descriptors"

NB! If the task contains unreferenced objects, these must be added one by one, using the Group Manager's Object browser to drag them into the right pane.



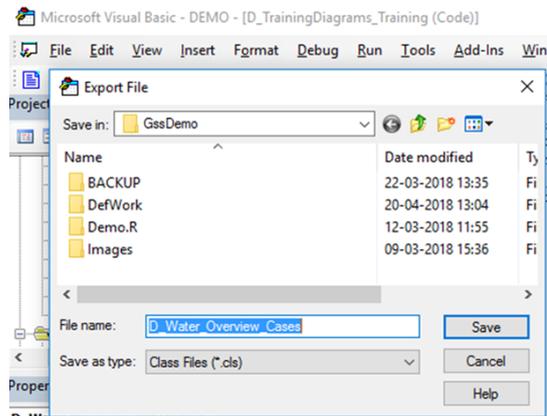
4. Finally, **export** the group to a .cgf file, and hand it to the project owner.

For more details on the Group Manager, kindly refer to the IGSS *Definition* help file's section "Bulk configuration (Multi-editing)".

- If you have VBA code in Modules, Class modules or Forms, these must be exported separately one by one using the VBA editor:

1. Open Visual Basic in the Definition menu: **Tools > Macro > Visual Basic Editor**

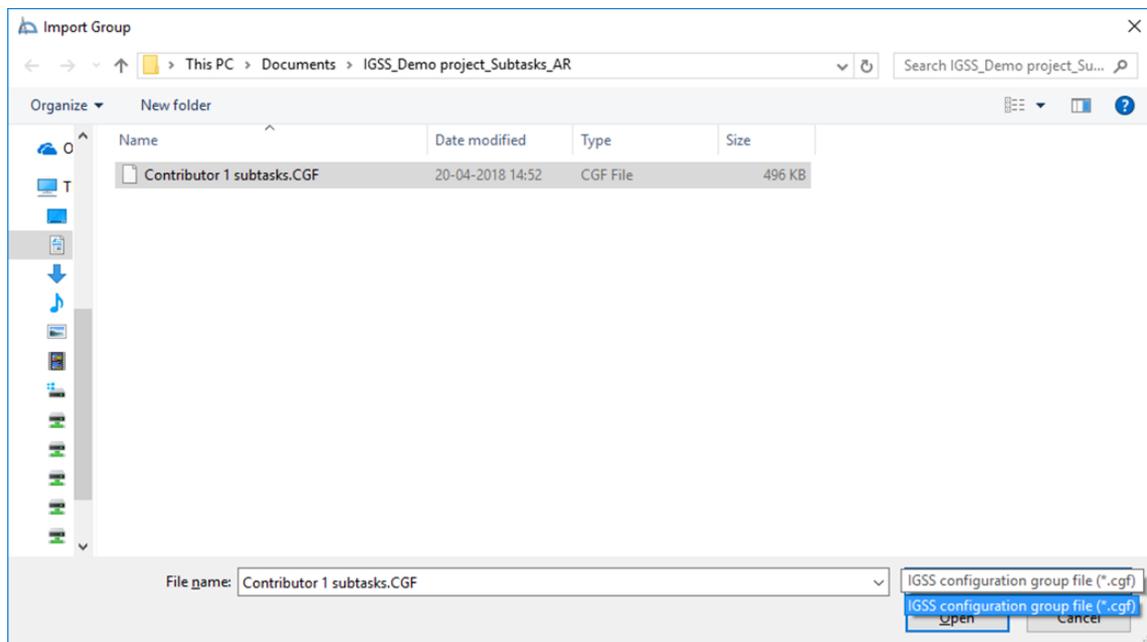
- In the VBA editor, mark the relevant module or form, select **File > Export** and save the file (various file formats). See screen dump next page.



2.6 Merge all files into base project (Project owner)



Now all the export files can be imported one by one into the final project in *Definition* using **File > Import**. Files exported using the Group Manager will also be imported.



Select "**References**" and "**Paste to new diagram**" for a smooth merge.

"Paste report formats" and "Paste event definitions" are optional.

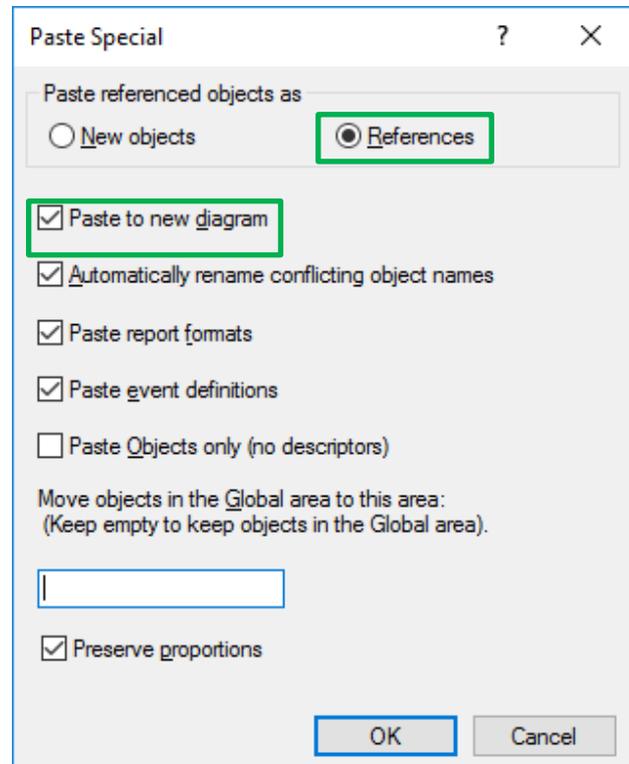
NB! VBA code from Modules, Class modules or Forms must be imported using the VBA editor's **File > Import**.

2.7 Post production (1 person)

A number of tasks can first be made when you have all the objects ready:

- Selection of "Name to menu" and/or "Description to menu" in Area, Diagram and Graph properties
- Overview diagrams and references between contributions
- Reports standard and Custom
- Maintenance jobs
- Events
- Notifier
- Dashboard
- Job scheduler
- Deployment

All these tasks must be done by one single system designer except for custom reports, which can be created during the contributors' subtasks and exported and imported into the final project as Excel files.



3 Adding new content to an existing project (Multiple designers)

If you want more system designers to add new content to an existing IGSS project, you can use the same principles and work processes as when building a new project from scratch.

NB! Bear in mind that the smallest project fragment you can distribute as a subtask is a diagram.

Project locking (available from IGSS version 13) offers an alternative way to distribute work to be done, not in parallel, but in a way that minimizes the risk of overwriting each other's changes. See section 4.

4 Project locking for adding new content (Multiple designers)

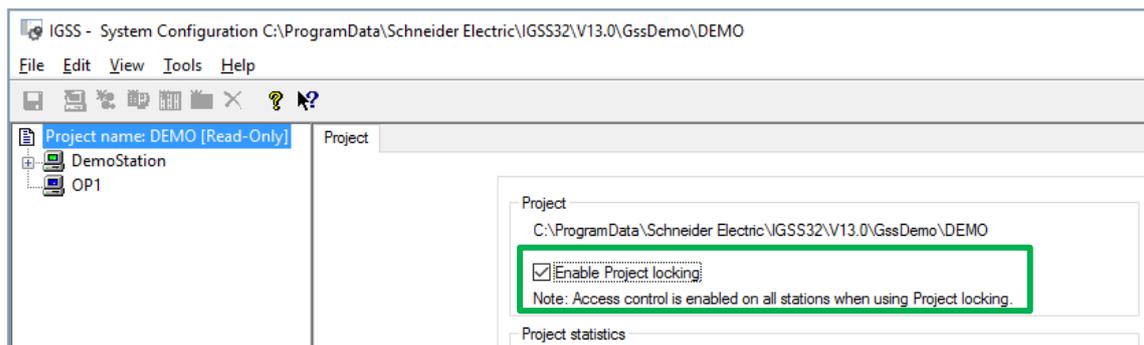
Locking a project, allowing only the person who locked it to access the *Definition* module, prevents the risk of having the project open for editing by several people at the same time – and thus several simultaneous versions of the same project. Project locking is available from IGSS version 13.



In the case that a small group of contributors need to work simultaneously on the project, e.g. when several system integrators cooperate on a project, and one system integrator is "at turn" and wants to allow several employees to work on the project, project locking can assist in assuring that the other system integrators do not interfere. Work distribution among the contributors at the system integrator in charge of the update is done as described in section 2.

Project locking – How it works

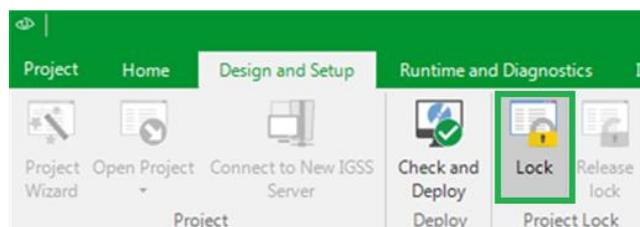
First, project locking must be enabled by the system administrator in IGSS Master.



Project locking requires that **Access control** has been enabled in *System Configuration*, and that users and user groups are defined in *User Administration*.

A system designer can first get access to the *Definition* module to edit the project when he or she has logged in and locked the project in *IGSS Master*.

When the project has been locked, the user having locked the project can edit the project from any station, server or operator station, where he/she has logged in.

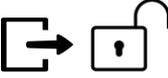


How to add new content with multiple contributors using project locking

If a system integrator wishes to allow several employees to work simultaneously on the same project, the project must first be locked before copying, and then unlocked before distribution. This can be done in different ways:

- If regular *IGSS user authentication* is used, the project can be locked and unlocked *off-site* by the user having originally locked it, and unlocked work files can be distributed.
- If *Windows user authentication* is used together with project locking, locking and unlocking the project *off-site* will have to be done on a station belonging to the same Windows domain as the site itself.

The workflow with multiple contributors and project locking is as follows:

1. **Lock:** A system integrator representative logs in on an operator station connected to the server, and locks the base project. If the system integrator does not have an operator station connected to the site, the project is copied from a station at the site. 
2. **Copy** the project to a station at the system integrator's premises and unlock it. If using *Windows authentication*, this must be done on a station on the same domain as the site. 
3. **Distribute** subtasks and unlocked project copies (project owner), create and export new content (multiple), merge and do postproduction (project owner) as described in section 2. 
4. **Copy the extended project** to an operator station connected to the server, replacing the base project:
The system integrator representative, who originally locked the project, logs in and copies the new project file to the project folder. 
5. **Put in operation:** Upload the extended project to the server by simply starting the project in *IGSS Master* on the operator station (NB! This requires that "Prompt for server connection and up/download of project" is selected in *IGSS System Configuration's* "Server & Backup tab for the operator station in question.) 

When asked, select the choice "Copy project files to the server". 

6. **Unlock** the project and log out.

5. Modifying existing content in a project (1 person)

This is always a one person task. Trying to come around it by deleting existing project components e.g. diagrams and importing new ones will break the relations to object history in log files and alarm files, and cannot be recommended.

Using project locking prevents others from changing the project while one person is making changes to the existing content in the *Definition* module.

6. More information

For more detailed instructions on how to create templates, import, export etc., kindly refer to the help function in your IGSS system and at www.igss.com > Support > Help and Manuals.

For trouble shooting, contact [IGSS Support](#).