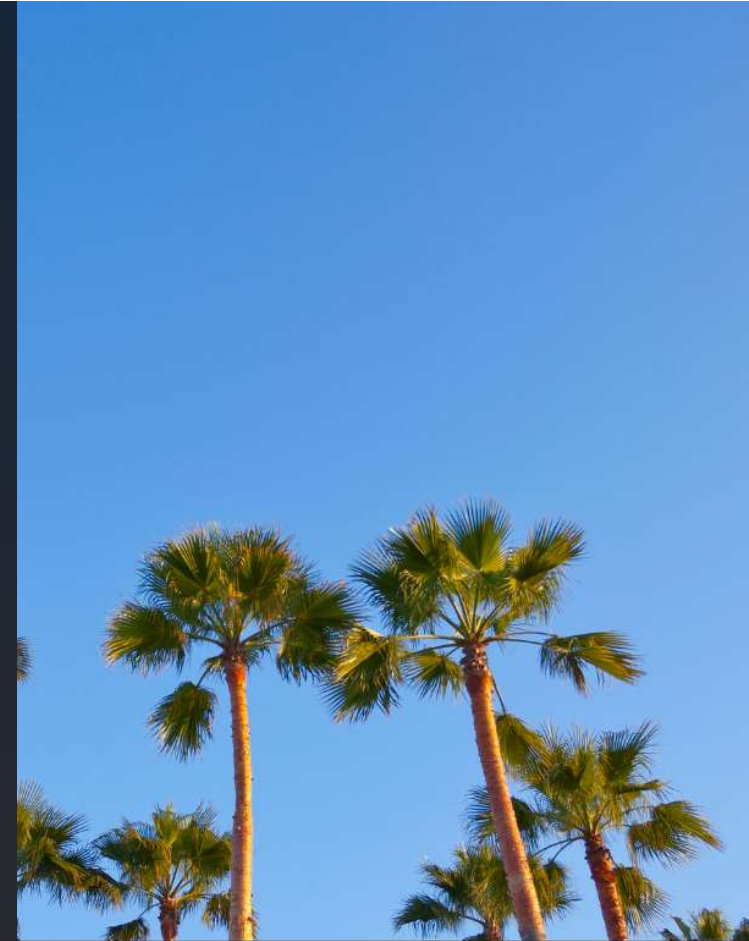


IEC 61850 IED Configuration Tool v2

- ▶ Eric Xu
- ▶ RTDS Technologies



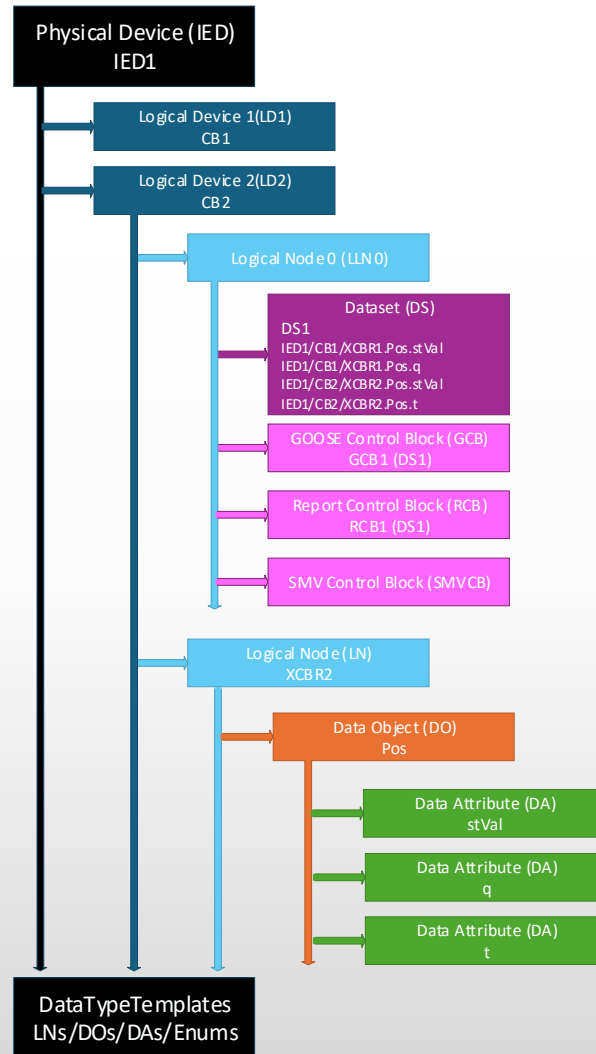
Outline

- Background
- Common ICT Use Cases
- ICTv1 – Challenges
- ICTv2 – Demonstration
- Summary

Background

The IED Configuration Tool (ICTv1) is used to configure GTNETx2-GSE and GTNETx2-MU for IEC 61850 applications. However, user and utility feedback has identified significant complexity and usability challenges, making the configuration workflow difficult and time-consuming.

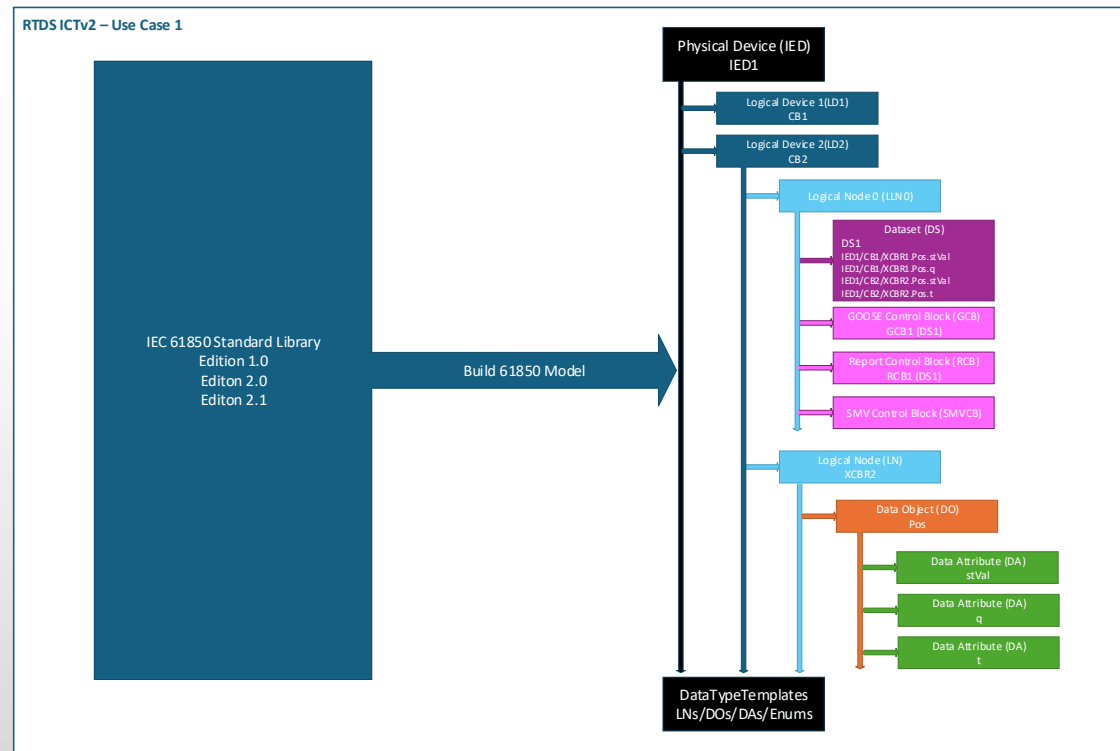




Simplified IEC 61850 Model Structure

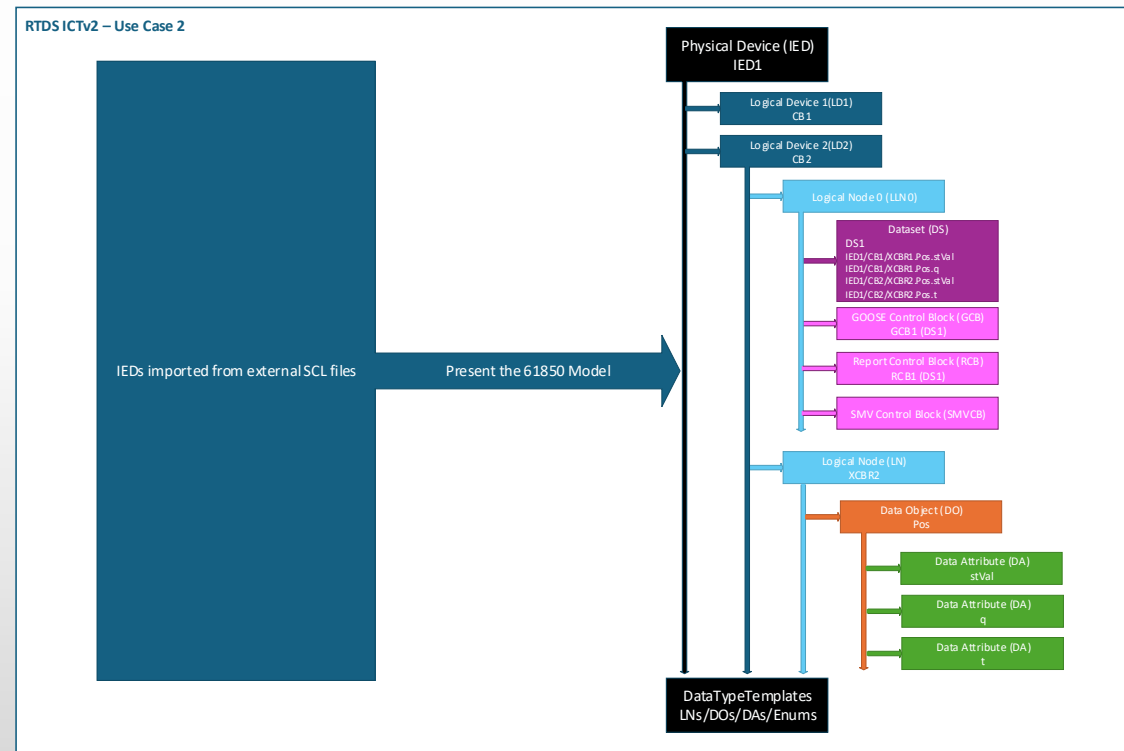
Overview - Use Case 1

Create an IEC 61850 data model from scratch using IEC 61850 standard data type templates within the tool



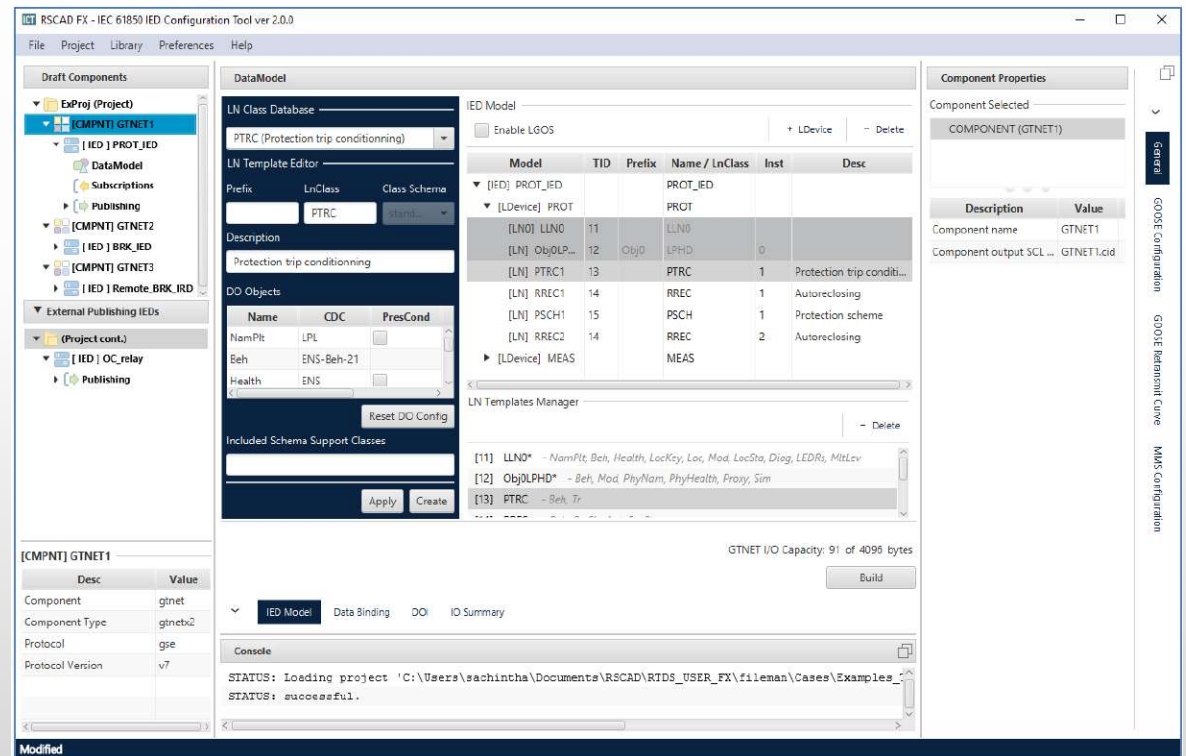
Overview - Use Case 2

Import a preconfigured SCL file to build and simulate the IEC 61850 IED data model defined in the file



ICTv1 Challenges

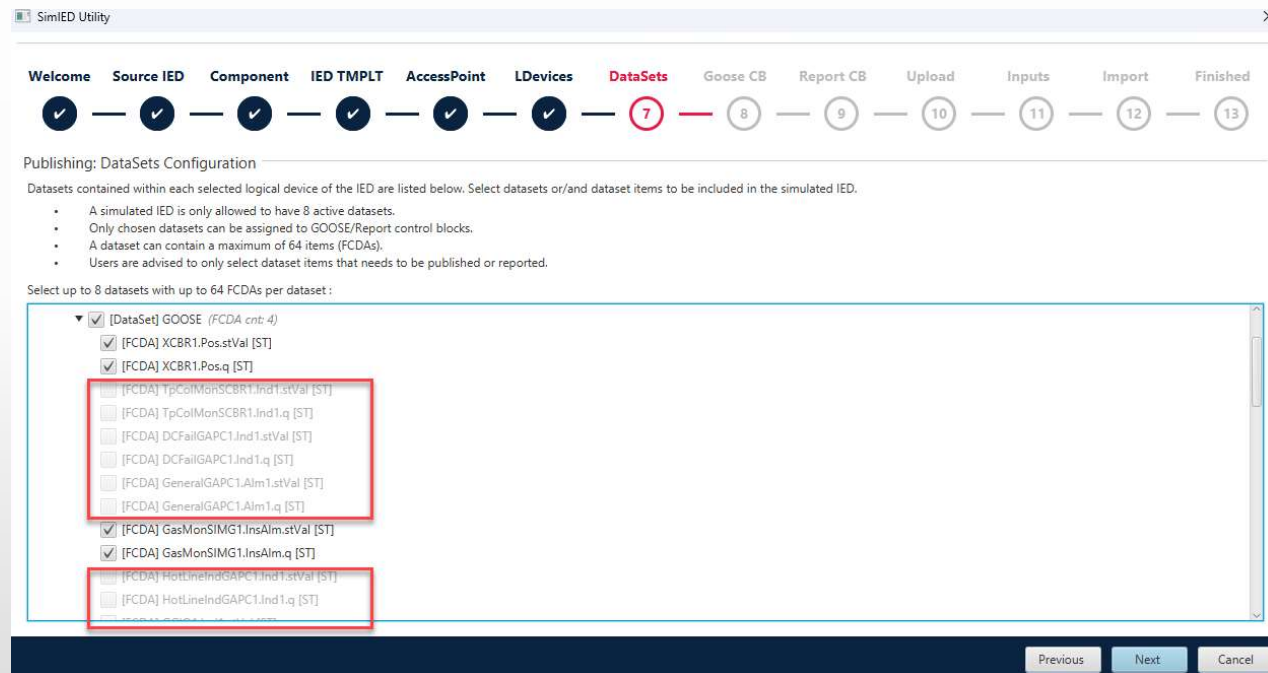
Excessive workflow complexity



ICTv1 Challenges

Limited Modeling Flexibility

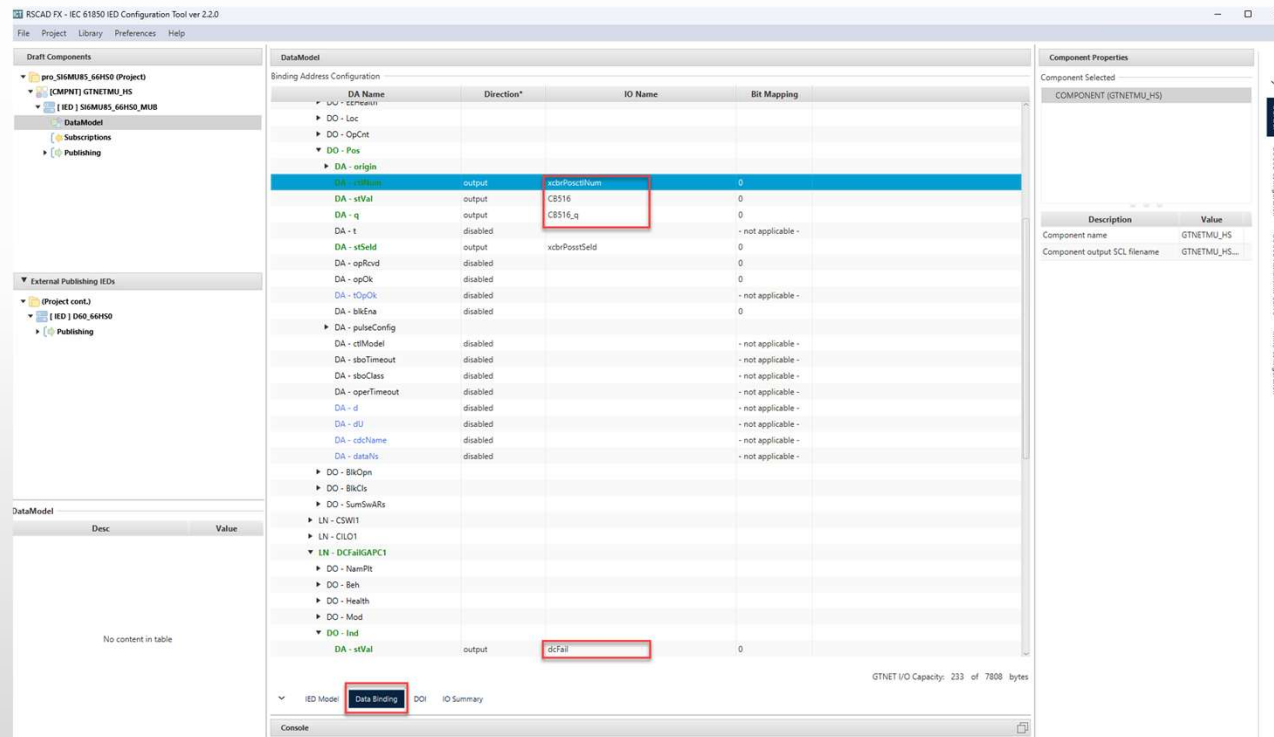
Vendor-specific Logical Nodes (LNs) and Data Objects (DOs) are not supported, limiting flexibility when modeling real-world IEDs



ICTv1 Challenges

I/O Naming and Mapping

RSCAD I/O names require configuration at multiple hierarchy levels, which increases setup complexity and the likelihood of mapping errors



ICTv1 Challenges

Scaling and Consistency

- As the size of a project grows, the configuration workflow in ICTv1 becomes increasingly difficult to scale.
- Repeating the same configuration patterns across multiple IEDs requires manual, error-prone duplication, making it hard to maintain consistency.
- As a result, inconsistencies are frequently discovered late in the project, leading to rework during integration or testing phases.

ICTv2 Objectives

- **Speed up IED configuration**, reducing engineering time and effort
- **Enable realistic utility-grade modeling** with support for vendor-specific IEC 61850 Logical Nodes and Data Objects
- **Improve interoperability and reuse** by supporting multiple SCL files within a single project
- **Reduce errors and improve consistency** through automated I/O naming
- **Increase productivity** with efficient bulk editing of I/O names
- **Minimize rework** with easy IED model comparison and fast configuration reloads
- **Enhance reliability and performance** through optimized CID file generation
- **Simplify project management and long-term support** by embedding ICT project files directly in the RSCAD RTFX file
- Deliver all enhancements without requiring firmware changes

IEC 61850

IED Configuration Tool v2

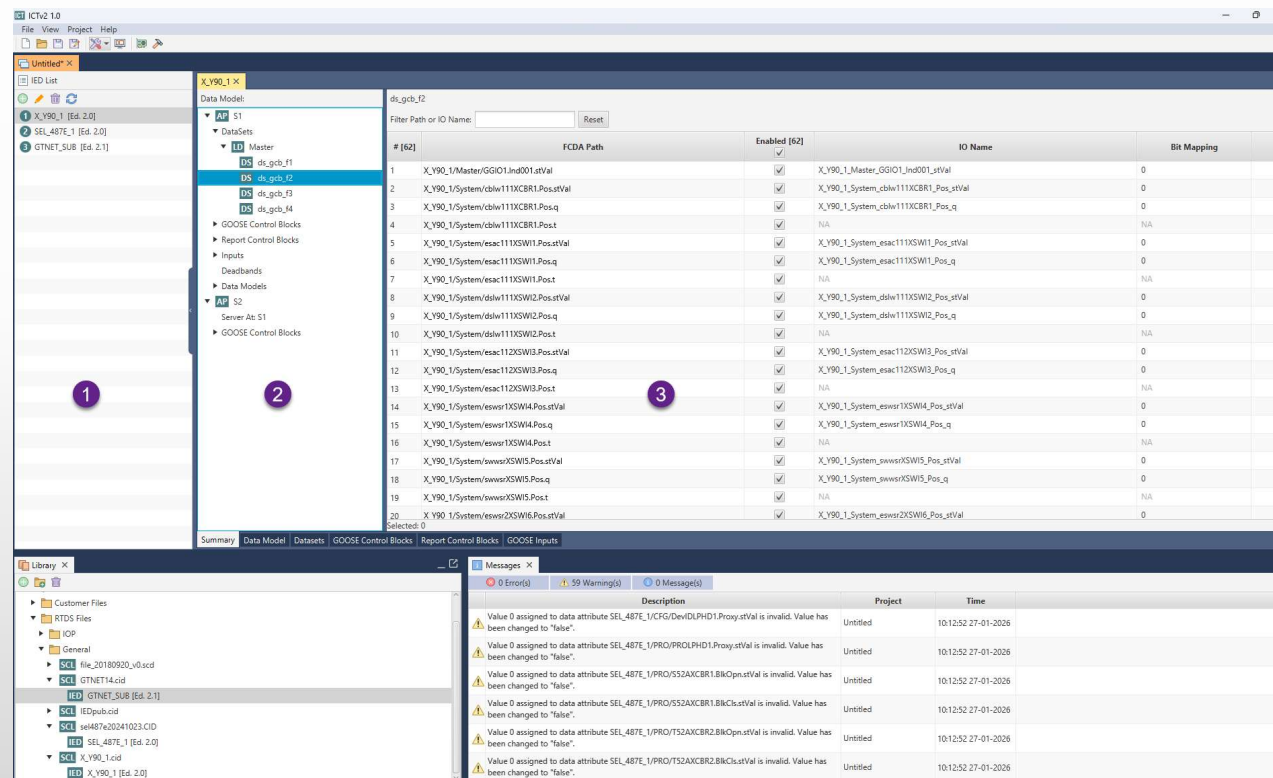


Your
world
in real
time.

User Interface

Modular, task-focused user interface

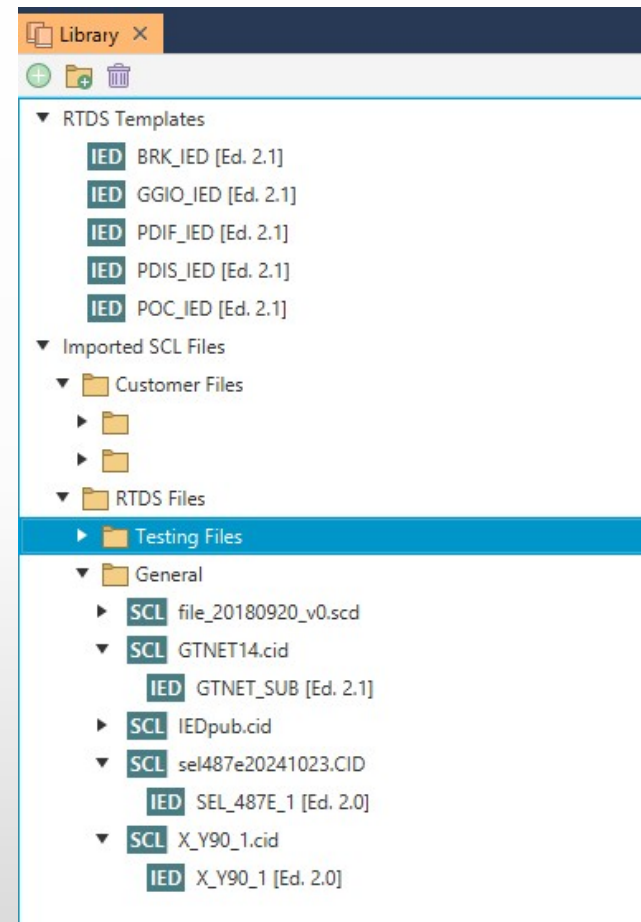
Fewer panels, clearer workflows, lower cognitive load



SCL Library

Scalable SCL library support

Multiple large SCL files, multi-IED projects



IED List

Centralized IED management

All project IEDs in one place



Summary Panel

Single-panel IED configuration

Datasets, control blocks, inputs, data model

The screenshot displays the IED configuration software interface. On the left, the 'IED List' shows three IEDs: X_Y90_1 [Ed. 2.0], SEL_487E_1 [Ed. 2.0], and GTNET_SUB [Ed. 2.1]. The main panel is titled 'X_Y90_1' and shows the 'Data Model' section. Under 'DataSets', the 'Master' block is expanded, showing 'ds_gcb_f1', 'ds_gcb_f2', 'ds_gcb_f3', and 'ds_gcb_f4'. The 'ds_gcb_f2' block is selected, and its configuration is shown in the right pane. The right pane has a 'Filter Path or IO Name' field and a 'Reset' button. Below this is a table with 6 columns: '# [62]', 'FCDA Path', 'Enabled [62]', 'IO Name', and 'Bit Mapping'. The table contains 16 rows of data. At the bottom, a 'Summary' tab is highlighted, and other tabs like 'Data Model', 'Datasets', 'GOOSE Control Blocks', 'Report Control Blocks', and 'GOOSE Inputs' are visible.

# [62]	FCDA Path	Enabled [62]	IO Name	Bit Mapping
1	X_Y90_1/Master/GGIO1.Ind001.stVal	☒	X_Y90_1_Master_GGIO1_Ind001_stVal	0
2	X_Y90_1/System/cblw111XCBR1.Pos.stVal	☒	X_Y90_1_System_cblw111XCBR1_Pos_stVal	0
3	X_Y90_1/System/cblw111XCBR1.Pos.q	☒	X_Y90_1_System_cblw111XCBR1_Pos_q	0
4	X_Y90_1/System/cblw111XCBR1.Pos.t	☒	NA	NA
5	X_Y90_1/System/esac111XSWI1.Pos.stVal	☒	X_Y90_1_System_esac111XSWI1_Pos_stVal	0
6	X_Y90_1/System/esac111XSWI1.Pos.q	☒	X_Y90_1_System_esac111XSWI1_Pos_q	0
7	X_Y90_1/System/esac111XSWI1.Pos.t	☒	NA	NA
8	X_Y90_1/System/dslw111XSWI2.Pos.stVal	☒	X_Y90_1_System_dslw111XSWI2_Pos_stVal	0
9	X_Y90_1/System/dslw111XSWI2.Pos.q	☒	X_Y90_1_System_dslw111XSWI2_Pos_q	0
10	X_Y90_1/System/dslw111XSWI2.Pos.t	☒	NA	NA
11	X_Y90_1/System/esac112XSWI3.Pos.stVal	☒	X_Y90_1_System_esac112XSWI3_Pos_stVal	0
12	X_Y90_1/System/esac112XSWI3.Pos.q	☒	X_Y90_1_System_esac112XSWI3_Pos_q	0
13	X_Y90_1/System/esac112XSWI3.Pos.t	☒	NA	NA
14	X_Y90_1/System/eswsr1XSWI4.Pos.stVal	☒	X_Y90_1_System_eswsr1XSWI4_Pos_stVal	0
15	X_Y90_1/System/eswsr1XSWI4.Pos.q	☒	X_Y90_1_System_eswsr1XSWI4_Pos_q	0
16	X_Y90_1/System/eswsr1XSWI4.Pos.t	☒	NA	NA

Dataset-Level I/O

Dataset-level I/O configuration

Automatic naming and bulk editing

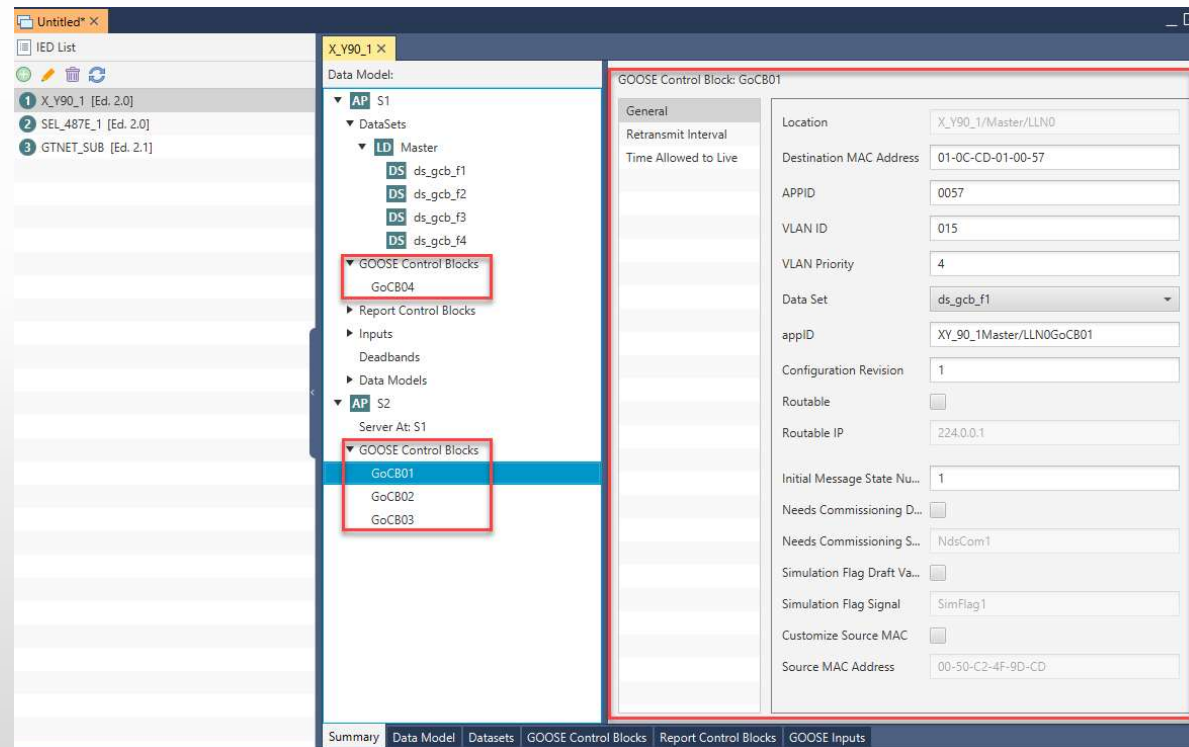
The screenshot shows the 'X_Y90_1' dataset configuration window. The 'Data Model' pane on the left shows a tree structure with 'AP S1' expanded, 'DataSets' expanded, 'LD Master' expanded, and 'DS ds_gcb_f1' and 'DS ds_gcb_f2' selected. The main table displays the configuration for 'ds_gcb_f2'. The table has columns: '# [62]', 'FCDA Path', 'Enabled [62]', 'IO Name', and 'Bit Mapping'. The 'IO Name' column shows values like 'X_Y90_1_Master_GGIO1_Ind001_stVal', 'gcb_f2_pos', 'X_Y90_1_System_cblw111XCBR1_Pos_q', 'qcb_f2_pos', 'X_Y90_1_System_esac111XSWI1_Pos_q', 'gcb_f2_pos', 'X_Y90_1_System_dslw111XSWI2_Pos_q', 'gcb_f2_pos', 'X_Y90_1_System_esac112XSWI3_Pos_q', 'gcb_f2_pos', and 'X_Y90_1_System_esw1XSWI4_Pos_q'. The 'Bit Mapping' column shows values like 0, NA, 2, 4, 6, 8. An 'Edit Multiple...' dialog box is open in the foreground, showing the 'IO Name' field with 'gcb_f2_pos' and the 'Bit Mapping' section with 'Don't Update' selected. The dialog also has an 'Enumeration' section with 'Enumerate' checked and 'Starting Value' set to 1. The 'Word Length' is 32 and 'Starting Bit' is 0.

# [62]	FCDA Path	Enabled [62]	IO Name	Bit Mapping
1	X_Y90_1/Master/GGIO1.Ind001.stVal	✓	X_Y90_1_Master_GGIO1_Ind001_stVal	0
	lw111XCBR1.Pos.stVal	✓	gcb_f2_pos	0
	lw111XCBR1.Pos.q	✓	X_Y90_1_System_cblw111XCBR1_Pos_q	0
	lw111XCBR1.Pos.t	✓	NA	NA
	ac111XSWI1.Pos.stVal	✓	qcb_f2_pos	2
	ac111XSWI1.Pos.q	✓	X_Y90_1_System_esac111XSWI1_Pos_q	0
	ac111XSWI1.Pos.t	✓	NA	NA
	lw111XSWI2.Pos.stVal	✓	gcb_f2_pos	4
	lw111XSWI2.Pos.q	✓	X_Y90_1_System_dslw111XSWI2_Pos_q	0
	lw111XSWI2.Pos.t	✓	NA	NA
	ac112XSWI3.Pos.stVal	✓	gcb_f2_pos	6
	ac112XSWI3.Pos.q	✓	X_Y90_1_System_esac112XSWI3_Pos_q	0
	ac112XSWI3.Pos.t	✓	NA	NA
	wsr1XSWI4.Pos.stVal	✓	gcb_f2_pos	8
	wsr1XSWI4.Pos.q	✓	X_Y90_1_System_esw1XSWI4_Pos_q	0
	wsr1XSWI4.Pos.t	✓	NA	NA

GCB / RCB

Unified GOOSE and report configuration

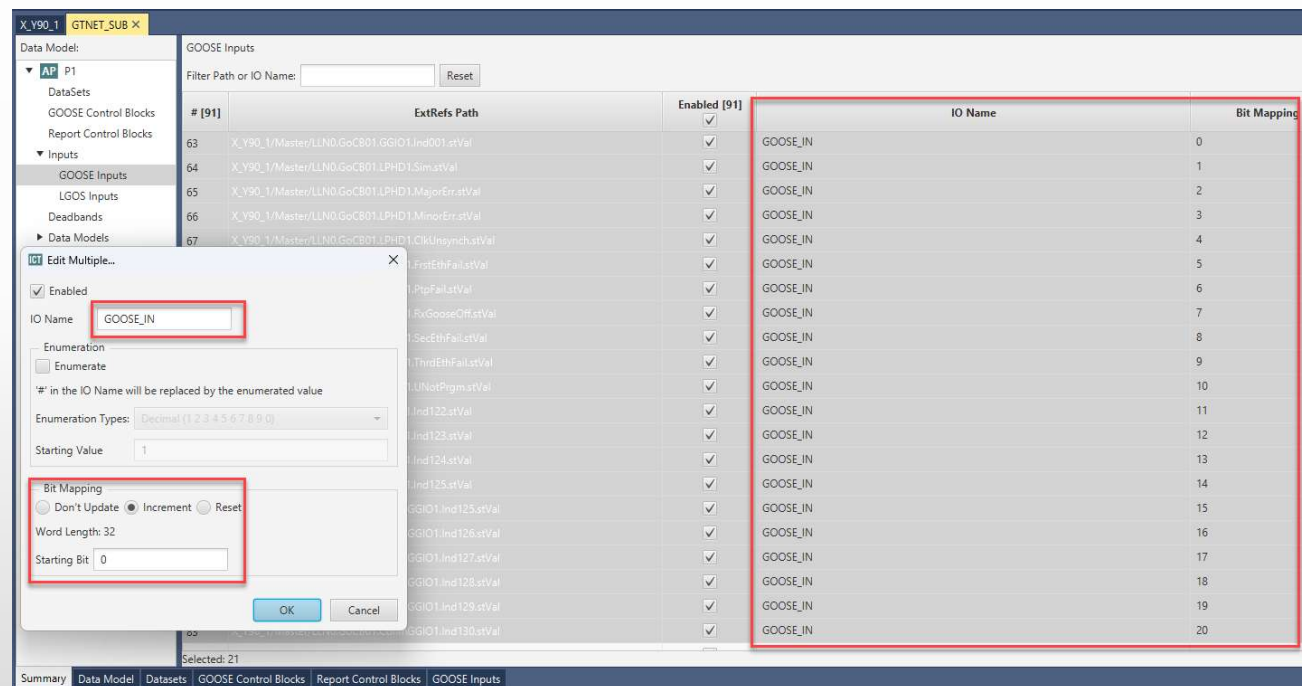
Communication and timing in one view



GOOSE Inputs

Efficient GOOSE input
management

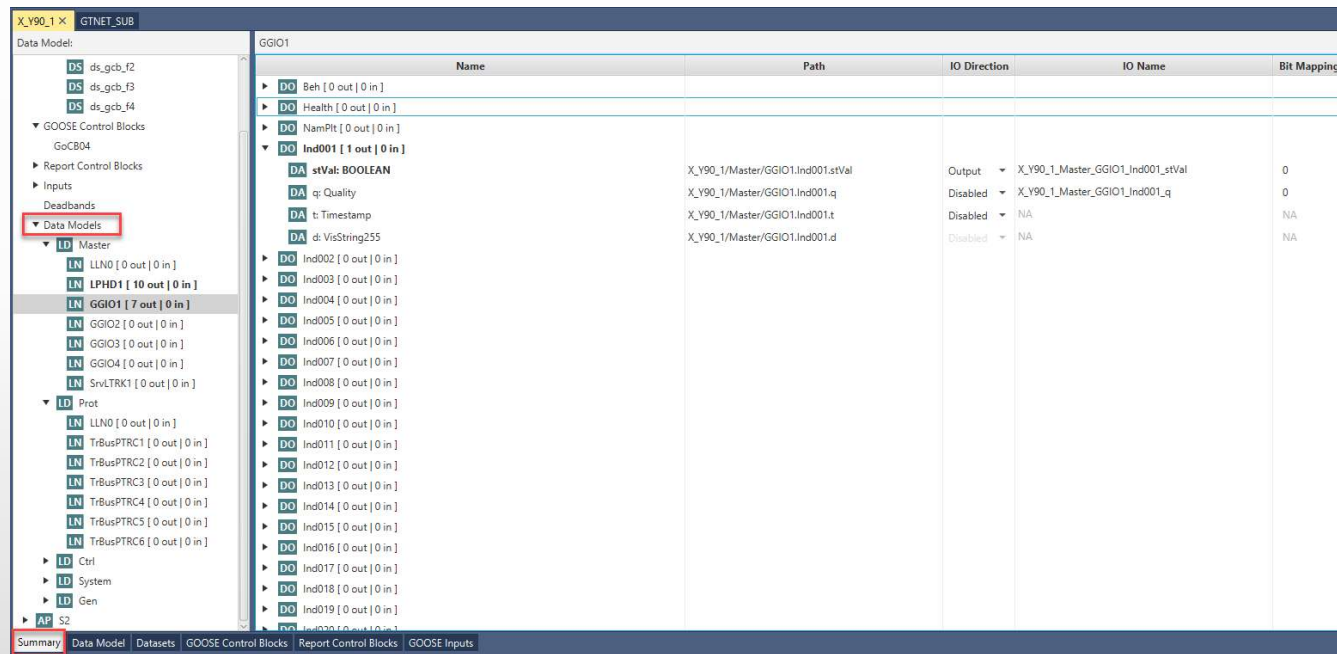
Automatic naming and batch edits



Data Model View

Complete IEC 61850 data model visibility

I/O naming synchronized across hierarchy

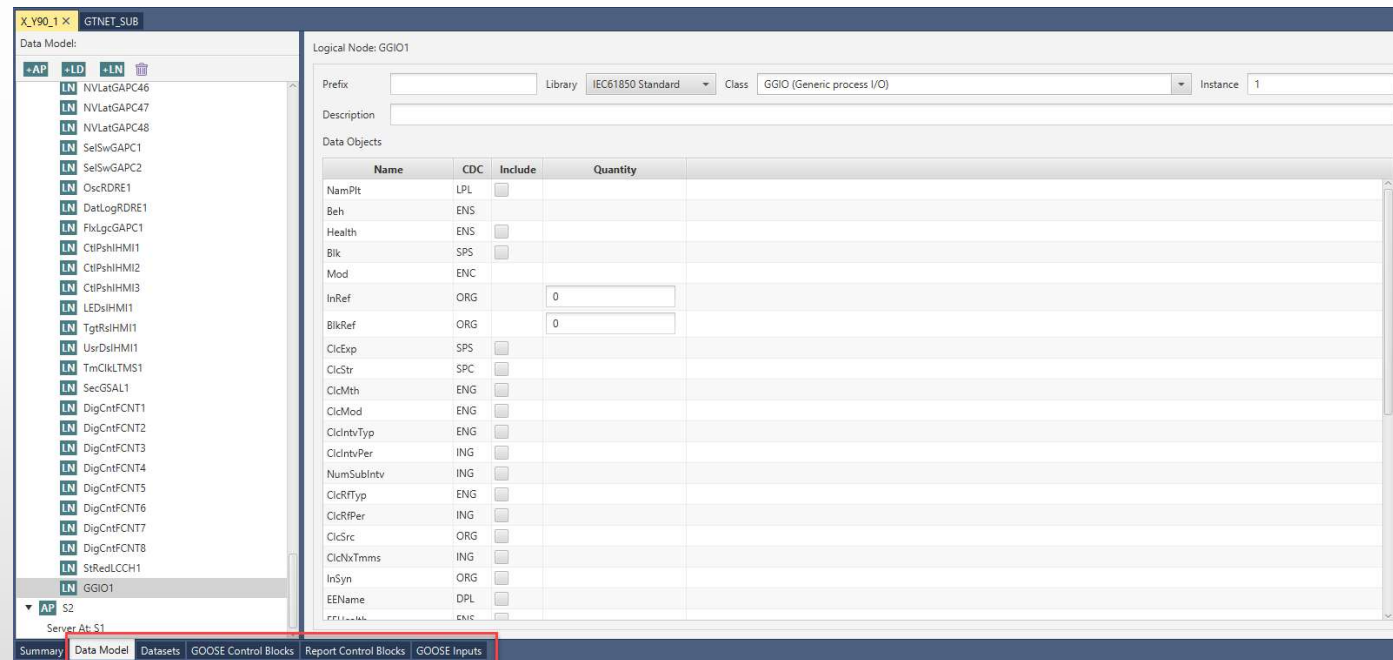


Name	Path	IO Direction	IO Name	Bit Mapping
Beh [0 out 0 in]				
Health [0 out 0 in]				
NamPrt [0 out 0 in]				
Ind001 [1 out 0 in]				
stVal: BOOLEAN	X_Y90_1/Master/GGIO1.Ind001.stVal	Output	X_Y90_1_Master_GGIO1_Ind001_stVal	0
q: Quality	X_Y90_1/Master/GGIO1.Ind001.q	Disabled	X_Y90_1_Master_GGIO1_Ind001_q	0
t: Timestamp	X_Y90_1/Master/GGIO1.Ind001.t	Disabled	NA	NA
d: VisString255	X_Y90_1/Master/GGIO1.Ind001.d	Disabled	NA	NA
Ind002 [0 out 0 in]				
Ind003 [0 out 0 in]				
Ind004 [0 out 0 in]				
Ind005 [0 out 0 in]				
Ind006 [0 out 0 in]				
Ind007 [0 out 0 in]				
Ind008 [0 out 0 in]				
Ind009 [0 out 0 in]				
Ind010 [0 out 0 in]				
Ind011 [0 out 0 in]				
Ind012 [0 out 0 in]				
Ind013 [0 out 0 in]				
Ind014 [0 out 0 in]				
Ind015 [0 out 0 in]				
Ind016 [0 out 0 in]				
Ind017 [0 out 0 in]				
Ind018 [0 out 0 in]				
Ind019 [0 out 0 in]				

Function Panels

Detailed control when needed

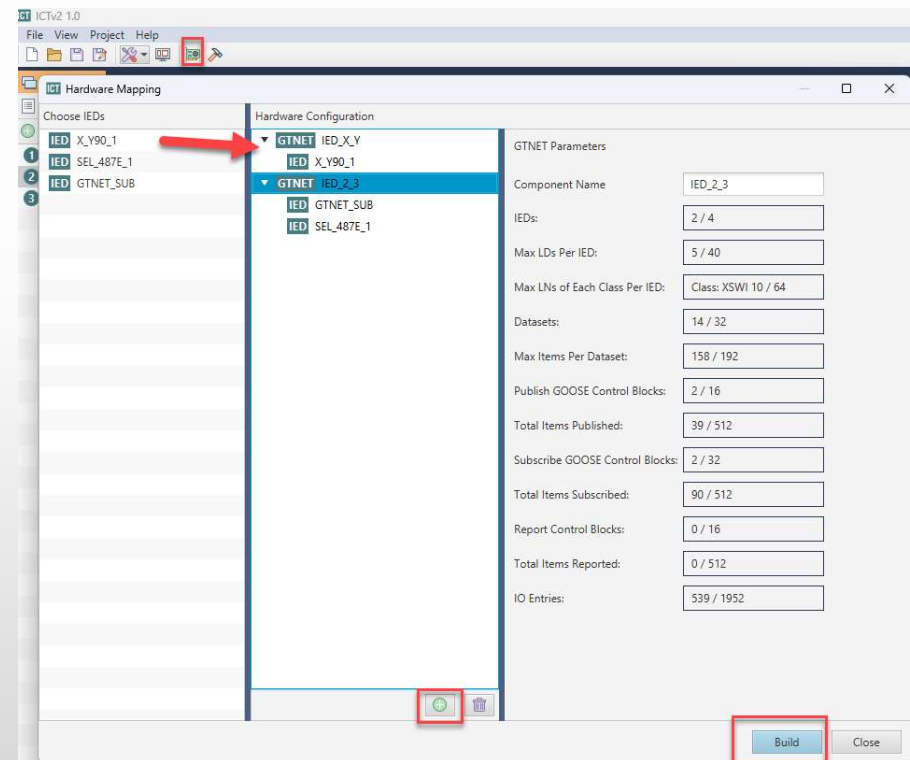
Advanced edits without cluttering the main workflow



Hardwire Mapping

Fast, intuitive GTNET mapping

Drag-and-drop with one-click build



Summary

ICTv2 simplifies large-scale IEC 61850 engineering

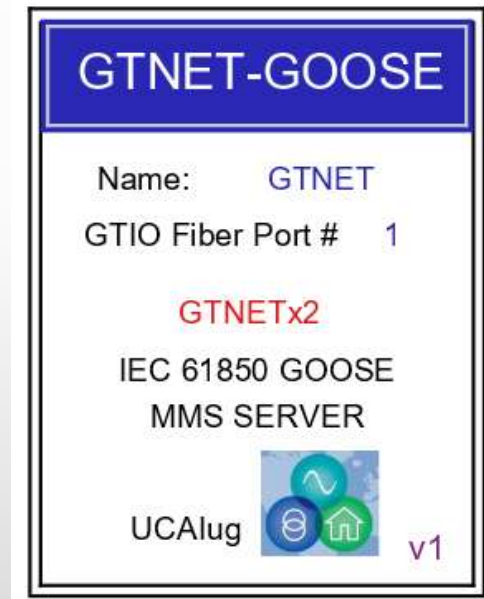
Faster setup, fewer errors, predictable workflows



Minimum Requirements

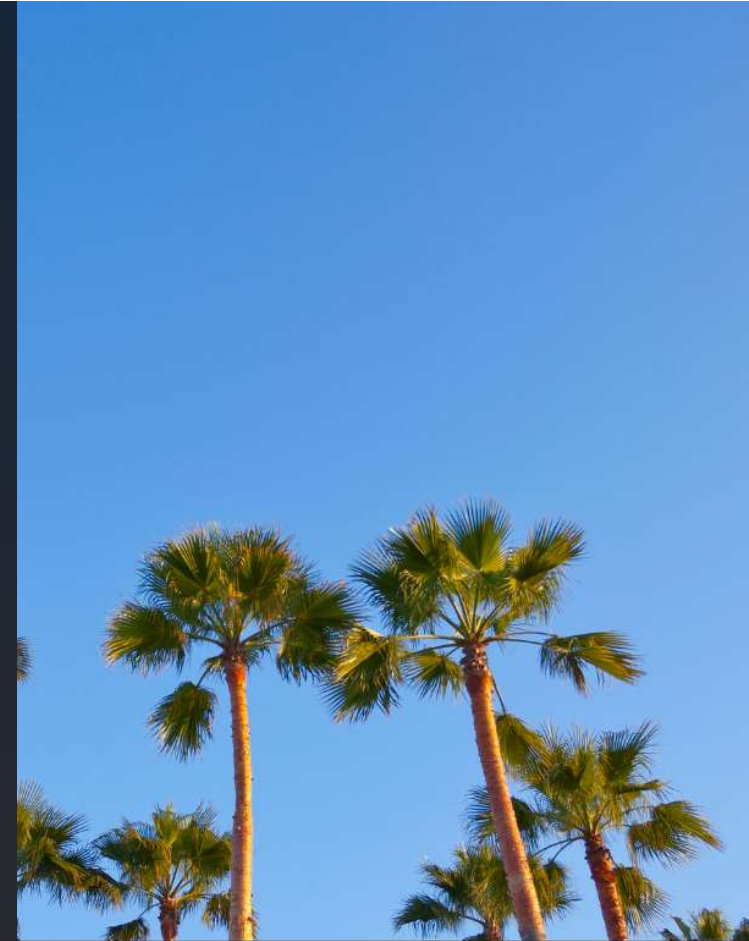
RSCAD FX 2.7

- GTNETx2-GOOSE-v1 RSCAD Component
- GTNETx2-GOOSE-v1 firmware
gtnet-goose_v1.11.bin
- NovaCor 1.0
WIF version 5.145 Build 7
RTFPGA version 0367b
- NovaCor 2.0
WIF version 6.145 Build 7
RTFPGA version 0423b



Thank You!

- ▶ Eric Xu
- ▶ RTDS Technologies



2026 APPLICATIONS & TECHNOLOGY CONFERENCE
IRWINDALE, CALIFORNIA, USA

