Sending Syslog Messages with a S7 CPU

Logging with SIMATIC CPU

Library description • June 2011

Applikations & Tools

Answers for industry.

SIEMENS

Siemens Industry Online Support

This article is taken from the Siemens Industry Online Support. The following link takes you directly to the download page of this document:

http://support.automation.siemens.com/WW/view/en/51929235

Caution
The functions and solutions described in this article confine themselves to the realization of the automation task predominantly. Please take into account furthermore that corresponding protective measures have to be taken up in the context of Industrial Security when connecting your equipment to other parts of the plant, the enterprise network or the Internet. Further information can be found under the Item-ID 50203404.

http://support.automation.siemens.com/WW/view/en/50203404

You can also actively use our Technical Forum from the Siemens Industry Online Support regarding this subject. Add your questions, suggestions and problems and discuss them together in our strong forum community:

http://www.siemens.com/forum-applications

SIEMENS

SIMATIC
Sending Syslog Messages with a S7 CPU

Library description

Warranty and Liability

Note

The library descriptions are not binding and do not claim to be exhaustive in terms of configuration and equipment as well as of any eventuality. The library descriptions do not represent customized solutions but shall give the user support in typical tasks. You are responsible for the correct operation of the described products. The present library descriptions do not relieve you from your obligation to provide professional knowledge in application, installation, operation and maintenance. By using the present library descriptions you recognize that, beyond the liability clause described above, we cannot be held responsible for any damage/claims that may arise. We reserve the right to make changes to the present library descriptions at any time without prior notice. The contents of the other documents will prevail in the case of discrepancies between the suggestions in the present library descriptions and other Siemens publications, such as catalogs.

We assume no responsibility for information contained in the present document.

Any claims against us - based on whatever legal reason - resulting from the use of the examples, information, programs, engineering and performance data etc., described in the present library description shall be excluded unless in cases where there is mandatory liability pursuant to the Product Liability Act for personal and property damages, in case of intent, gross negligence, personal injury, bodily harm or damage to health, owing to the acceptance of a guarantee for the quality of an object, fraudulent concealment of a defect or breach of essential contractual duties. However, claims for damages arising from breach of essential contractual duties shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for personal injury, bodily harm or damage to health. The above provisions do not imply a change in the burden of proof to your detriment.

Table of Contents

1 Task definition and solution

Introduction

The syslog protocol is in fact one of the simplest loggers. An application uses syslog to transfer messages, warnings or error statuses to a remote server and, if required, file the messages in a database.

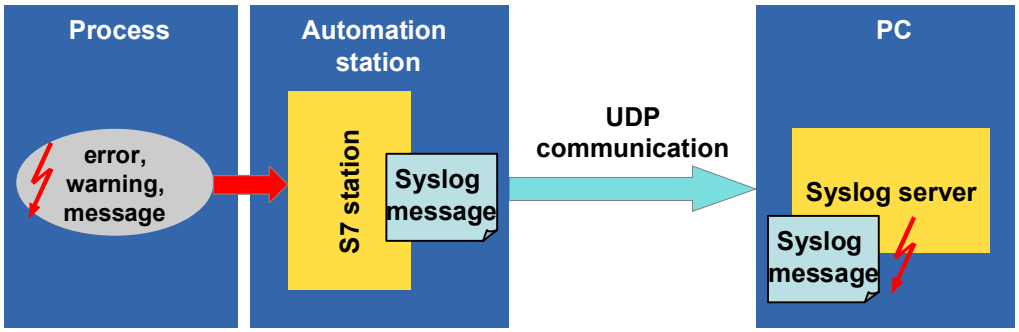
To provide this functionality to the SIMATIC CPUs too, a block for a library is to be programmed which emulates the syslog protocol and sends messages to a server via open communication.

1.1 Task definition

The task at a glance

The figure below provides an overview of the task.

Figure 1-1



Description of the task

The objectives below shall be reached:

- Emulating the syslog protocol in the user program
- Establishing an open communication between control system and PC
- Sending syslog messages to a syslog server
- Configuring a database connection to the syslog server

A function block to be created for such purpose shall take over and execute the first three tasks on the basis of little external information.

Requisites of the block

Table 1-1

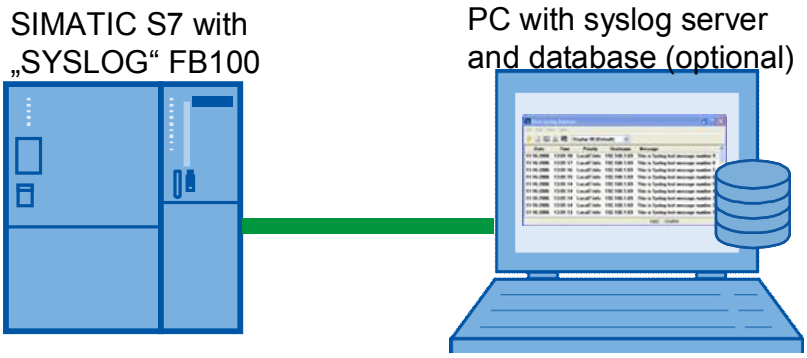
Requisite	Explanation
Executable on all control systems.	Programming language must be understood by all current and future control systems.
Independency	Preferably be self-sustaining and only be fed with the absolutely necessary.
Able to have parameters assigned.	Customized configuration by the user of the syslog messages based on the display level and the display text.

1.2 Solution

Schematic layout

The following scheme shows the most important components of the solution:

Figure 1-2



Description

The automation solution includes two centralized communication partners:

- One PC acting as receiver of the syslog messages (server).
- Control system acting as sender of the syslog messages (client).

The control system periodically calls the SYSLOG FB100 and creates and sends a syslog message if need be.

The PC runs the syslog server and a SQL database, as an option, which receive, process and display the massage.

Applicability

Supports all current PROFINET CPU and CP of the SIMATIC product range:

- CPU 31x-2 PN/DP from FirmWare version 2.4 up
- CPU 41x-3 PN/DP
- ET 200S PN-CPU
- S7-300 Ethernet CPs
- S7-400 Ethernet CPs

STEP 7 V5.5 SP2 is used as configuration software for the PLC.

In principal any commercially available server can be used as syslog server.

1.3 Scenarios

"SYSLOG generation" scenario

The SYSLOG block is in the heart of this scenario. This block performs the following tasks:

- Establishing an UDP connection to a remote syslog server via open communication blocks.
- Generating and emulating the syslog protocol with the help of the information transferred by the user.
- Sending the syslog message.

"Display and archiving" scenario

One syslog server assisted by a SQL database, if need be, handle the displaying and archiving of generated syslog messages.

Any syslog message can be given a priority level and is displayed color-encoded in the syslog server's display window.

Figure 1-3

!	Date	Time	Priority	Hostname	Message
	07-21-2011	09:24:44	Auth.Alert	192.168.22.10	Unauthorizised user
	07-21-2011	09:21:47	System0.Info	192.168.22.10	Warmstart of the CPU
	07-21-2011	09:19:12	Local0.Critical	192.168.22.10	Temperature too high

2 Library overview

What do you obtain?

The present document describes the SYSLOG blocks library. This blocks library includes tested code with well-defined interfaces. You can use them as basis for your projected task.

The library provides three S7 programs folders:

- **SYSLOG_PN** when using PROFINET CPU and the integrated PROFINET interface.
- **SYSLOG_CP300/ CP400** when using standard CPU and one CP as communication block.

Both S7 programs have the same functionality. Owing to different interfaces (integrated PROFINET interface and external communication block) the program is executed in different manner.

The gist of this document is to describe:

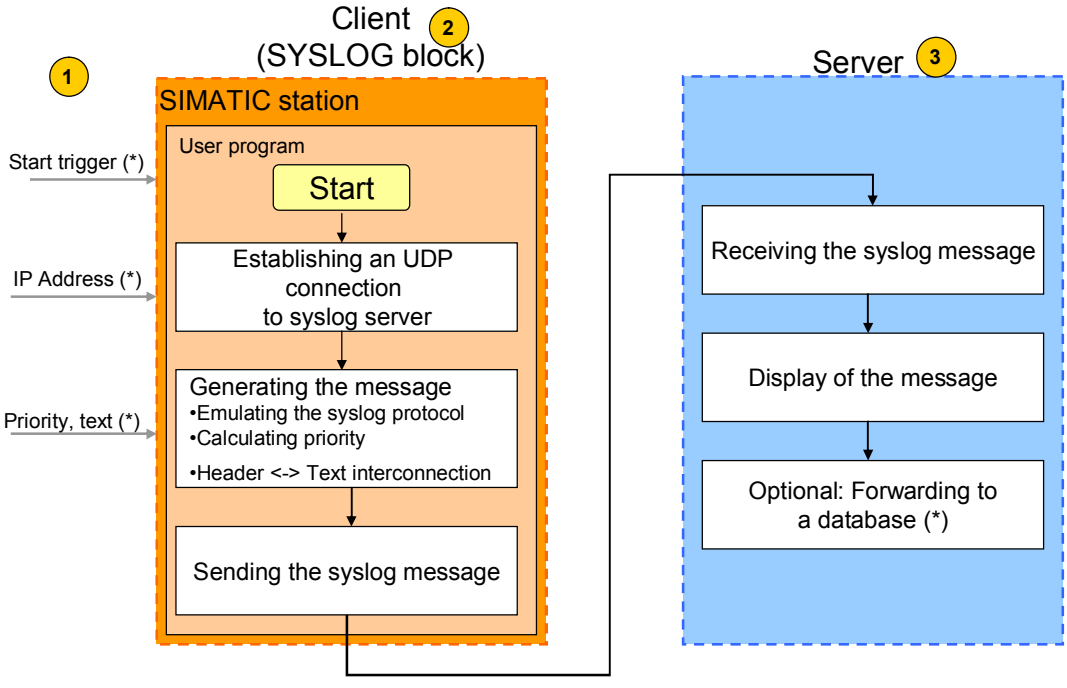
- all blocks pertaining to the library
- the functionality provided by these blocks.

The present documentation furthermore illustrates possible applications and the included step-by-step instructions help you to integrate the library into your STEP 7 project.

2.1 Workflow

The figure below gives a schematic overview of the SYSLOG library's functionality:

Figure 2-1



(*) Is to be configured / assigned with parameters by the user

Table 2-1

No.	Action	Note
1.	Assign parameters to syslog client.	The parameters assigned to the SYSLOG block (client) provide all information required to generate and send syslog messages.
2.	Client starts processing upon trigger signal.	The syslog message is being generated and sent.
3.	The server receives the message, displays it and forwards it to a database for archiving, if need be.	The forwarding must be explicitly activated in syslog server and a database is to be created.

2.2 Program blocks

Overview

The SYSLOG library features one centralized, customized function block and several standard blocks of the SIMATIC standard S7 and IEC library.

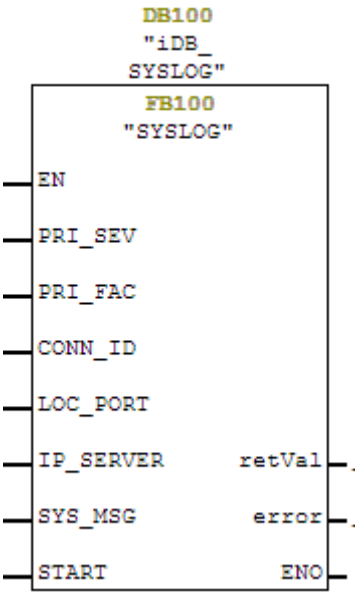
Developed as state machine the "Syslog" FB100 core block handles the generation and sending of the syslog message.

FB100 works with the standard blocks of the SIMATIC library for such purpose.

Centralized SYSLOG block (FB100)

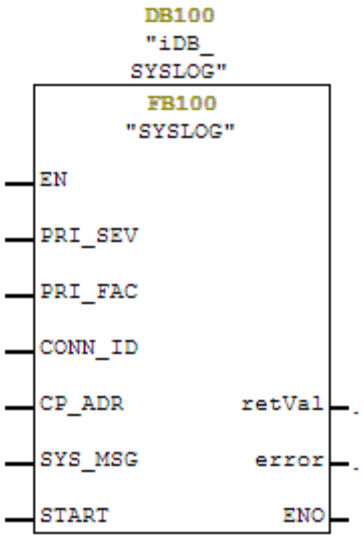
The chart below shows the FB100 for the integrated PROFINET interface (S7 program folder SYSLOG_PN):

Figure 2-2



The chart below shows the FB100 when using a communication processor (S7 program folder SYSLOG_CP):

Figure 2-3



The parameters have the following meaning:

Table 2-2

Type	Variable	Data type	Default value	Meaning
Input	PRI_SEV	INT	0	Defines the severity for the priority (Severity field)
	PRI_FAC	INT	0	Defines the provenience for the priority (Facility field)
	CONN_ID	BYTE	B#16#1	Defines the ConnectionID for the communication.
	LOC_PORT	WORD	W#16#7D00	Local port for the UDP connection (only available when using the integrated PN interface).
	IP_SERVER	DWORD	DW#16#000000	Defines the syslog server's IP address (only available when using the integrated PN interface).
	CP_ADR	INT	256	Start address of the CP (only available when using CP)
	SYS_MSG	ANY		ANY pointer to the message text. 230 characters maximum!
	START	BOOL	0	Triggers transfer of the syslog message.
Output	retVal	DWORD	0	Status message of the block. The first word reflects the error source. The second word reads out the error message of the internally called block.

Type	Variable	Data type	Default value	Meaning
	error	BOOL	0	Is set if an error has occurred during processing.

Auxiliary blocks of the SIMATIC library

The centralized FB100 block internally calls further blocks for supporting the generation and sending of syslog messages. The following blocks are concerned:

Table 2-3

Auxiliary blocks	Integrated in program folder	Description
TCON (FB65) TUSEND (FB67) TDISCON (FB66)	SYSLOG_PN	Open communication blocks to establish UDP connections.
AG_SEND (FC5)	SYSLOG_CPx00	Communication block to transfer data to an Ethernet CP for transmission via configured connection.
CONCAT (FC2) DELETE (FC4) INSERT (FC17) I_STRNG (FC16)	SYSLOG_PN and SYSLOG_CPx00	IEC functions to edit STRING variables.
TON (SFB4) BLKMOV (SFC20)	SYSLOG_PN and SYSLOG_CPx00	System functions for the watchdog timer and to copy arrays.
RDSYSST (SFC51)	SYSLOG_PN	System function to read data records.

2.3 Hardware and software requirements

Requirements for the library

To make use of the full functionality of the library described here the hardware and software requirements listed below must be met.

Hardware

The blocks of the SYSLOG library can be integrated in all STEP 7 projects whatever hardware is used.

The SYSLOG_PN S7 program folder is designed for the use of one PROFINET CPU with integrated Ethernet interface. It supports all current PROFINET CPU and CP of the SIMATIC product range:

- CPU 31x-2 PN/DP from Firmware version 2.4 up
- CPU 41x-3 PN/DP
- ET 200S PN-CPU

The SYSLOG_CP300 S7 program folder is designed for the use of standard CPU with CP343-1 communication processor.

It supports all current Ethernet CP of the SIMATIC product range:

- CP343-1/ CP343-1 Advanced
- CP343-1 Lean

The SYSLOG_CP400 S7 program folder is designed for the use of standard CPU with CP443-1 communication processor.

It supports all current Ethernet CP of the SIMATIC product range:

- CP443-1 Advanced
- CP443-1

Software

STEP 7 V5.5 SP2 is used as configuration software.

Any commercialized software can be used for the syslog server.

If you also wish a connection to the SQL database, the syslog server must support this function.

The software below has been used to create the present documentation:

- KIWI Syslog Server Version 9.2
- MySQL-Server V5.5
- SQL Manager for MySQL Server
- mysql-connector-odbc-5.1.8-win32.msi

2.4 Library resources and performance data

What will you learn here?

The overview below shows the assignment to RAM of the SYSLOG library's blocks.

Library resources

The table below gives the size of the program blocks or instance data blocks in RAM:

Table 2-4

No.	Block	Size	Note
1.	SYSLOG (FB100)	2954 Byte	For the integrated PROFINET interface
2.	Instance data block for FB100	754 Byte	
3.	Local data (FB100)	356 Byte	
4.	SYSLOG (FB100)	1894 Byte	When using a communication processor
5.	Instance data block for FB100	610 Byte	
6.	Local data (FB100)	316 Byte	

Watchdog timer

The watchdog timer prevents the function block from being stuck in an infinite loop upon an error. The default timer is set to be 400ms.

Syslog message

Theoretically, the syslog message text may comprise as many as 1024 bytes. For this library, however, the threshold was set to 235 characters (header + text) because the AG_SEND communication block used in connection with a S7-400 CP provides a limited data range.

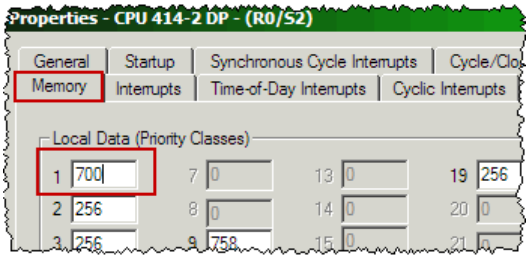
As the header may include up to 5 characters the message text may have 230 characters maximum (230 characters + 5 characters = 235 characters).

Local data

The SYSLOG block needs local data to process the internal program. The number of local data required can be taken from Table 2-4.

In case of a S7-400 station, pay attention that the local data size of S7-400 CPU priority classes is sufficiently dimensioned for the SYSLOG block.

Figure 2-4



3 The SYSLOG function block in detail

What will you learn here?

This chapter describes in detail the core functionality and the internal operation workflows of the SYSLOG library block.

Note

The reader shall have adequate knowledge of the syslog protocol to understand this chapter.

Further information on the syslog protocol can be found in Appendix (chapter 5 Syslog protocol).

3.1 Emulating the syslog protocol

As mentioned above the syslog protocol is of very simple structure. It merely consists of a header and the message text.

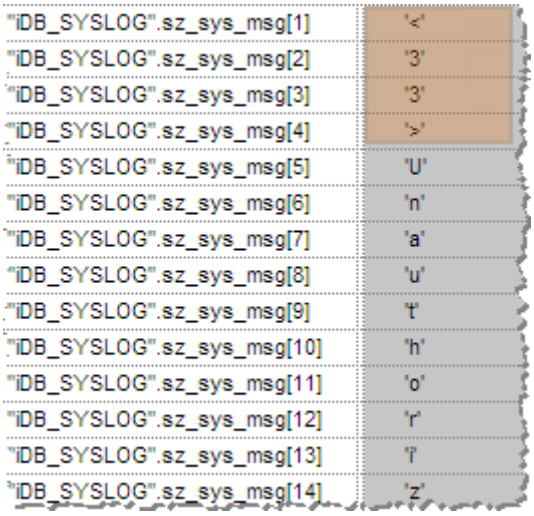
To configure the header the block must calculate the priority of the syslog message from the transferred Severity and Facility values and save it in the required format (< [priority value] >). The formula below is used for calculation:

Priority value = Facility value * 8 + Severity.

The result is saved number by number as CHAR in ASCII format (one byte per number is needed).

The detail below taken from the variables table shows how to proceed:

Figure 3-1



3.2 Establishing an UDP connection

3.2.1 T blocks for a PROFINET CPU

General information

Syslog messages are sent over network via UDP / IP. The open communication blocks TCON, T_USEND and T_DISCON are used for UDP communication. The table below explains which tasks these blocks have:

Table 3-1

Block	Task
TCON	The communication partner calls FB 65 "TCON" to configure the local port. Therefore, a connection between user program and the operating system's communication layer is established. As UDP is without connection, no connection is established to the remote partner.
TUSEND	FB 67 "TUSEND" sends data to a remote partner via UDP. The address is stored in the configuration UDT.
TDISCON	FB 66 "TDISCON" disconnects the local port, this means, terminates the connection between user program and operating system's communication layer.

Assigning parameters to T blocks

Normally, the "Open Communication Wizard" is used for simple parameter assignment to the T blocks.

As independency is one item of the block's requisites list, the user himself must assign parameters to the T blocks in the user program.

Two data structures (UDT) are required to assign parameters to T blocks:

1. Parameterizations for the connection build-up
2. Configuration data for sending

The parameterizations for the T_CON block must include the following:

Table 3-2

Parameters	Data type	Description
block_length	WORD	UDT length (64 bytes fixed).
id	WORD	Connection ID
connection_type	BYTE	Protocol variant (UDP: B#16#13)
active_est	BOOL	Type of connection build-up (UDP: False)
local_device_id	BYTE	Interface identifier, depending on the CPU type
local_tsap_id_len	BYTE	Length of the parameter local_tsap_id (2 bytes).
rem_subnet_id_len	BYTE	Not used; 0 must be assigned here.
rem_staddr_len	BYTE	Not used; 0 must be assigned here.
rem_tsap_id_len	BYTE	Not used; 0 must be assigned here.
next_staddr_len	BYTE	Not used; 0 must be assigned here.
local_tsap_id	ARRAY[1..16] of BYTE	Local port number

Parameters	Data type	Description
rem_subnet_id	ARRAY[1..16] of BYTE	Not used; 0 must be assigned here.
rem_staddr	ARRAY[1..16] of BYTE	Not used; 0 must be assigned here.
rem_tsap_id	ARRAY[1..16] of BYTE	Not used; 0 must be assigned here.

The configuration data for T_USEND have the following definitions:

Table 3-3

Parameters	Data type	Description
rem_ip_addr	ARRAY[1.0.4] of BYTE	IP address of the remote partner
rem_port_nr	ARRAY[1..2] of BYTE	Port address of the remote partner
spare	ARRAY[1..2] of BYTE	Not used; 0 must be assigned here.

Note The structure of the configuration data / parameterization blocks must not be changed.

The remote partner's IP address is predefined by the IP SERVER input parameter.

The syslog specification defines the remote port number (514) and the connection type (UDP).

The user is free to choose the local port number and the connection ID.

The UDT block lengths, the connection type and the length of the local tsap are default.

Solely the interface identifier must be read out of the used block by the user program.

Reading out the interface identifier

The system status list (SSL) W#16#011C and the index W#16#7 provide the interface identification. The partial SSL helps read out the CPU type.

Comparing the strings or searching for an explicit numerical sequence (e.g. 317) the user program helps automatically determine the CPU type and encode it for the T blocks.

3.2.2 Configured connection for Ethernet CP

General information

T blocks are not appropriate when an Ethernet CP is used. In this case an UDP connection must be established explicitly via NetPro.

The CPU uses the AG_SEND block to communicate to the CP all data to be transported by this connection.

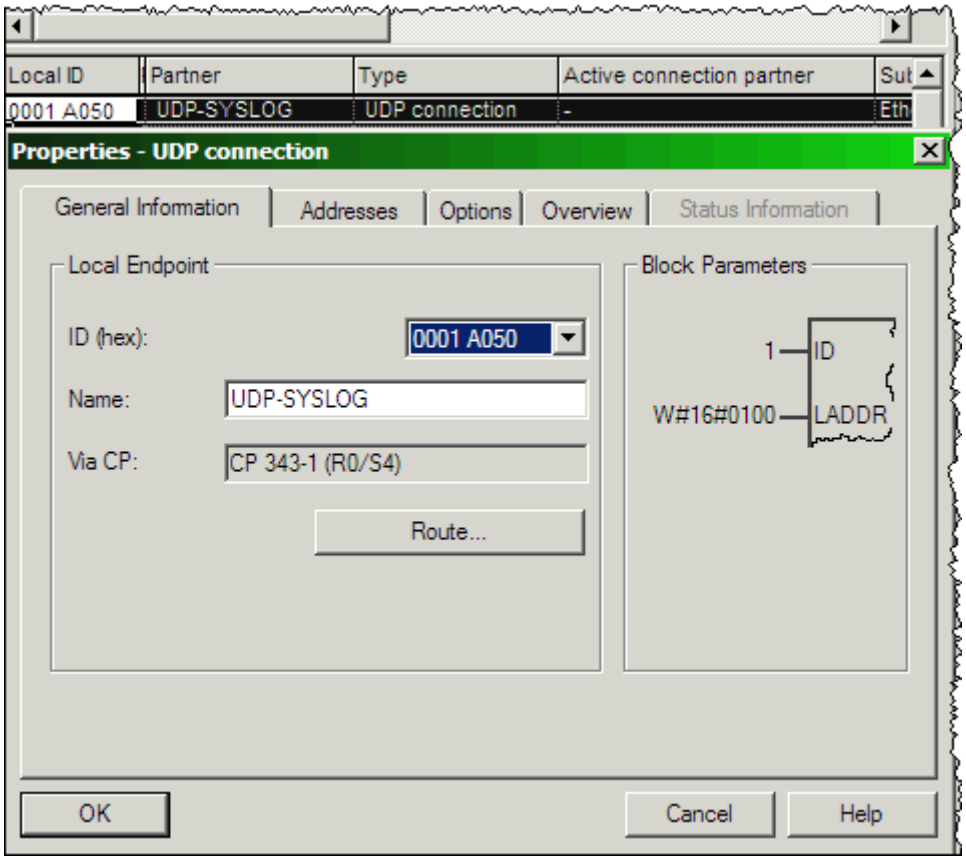
Configuring an UDP connection in NetPro

The use of an Ethernet CP as communication interface requires that the necessary connection be configured. In NetPro, though, an unspecified UDP connection is to be created since the syslog message is transmitted UDP / IP-based.

To configure this UDP connection the IP address of the remote partner (the syslog server's IP address) and the remote port address (514 for syslog) are required.

Any connection is assigned a unique ID which must be indicated when the communication blocks are called.

Figure 3-2



Note The S7-400 / S7 300 series' CP need different versions of AG_SEND (FC5); therefore, separate program folders were created in the library.

3.3 Error and status display

If an error occurs the SYSLOG function block (FB100) displays an error code via the retVal output parameter.

The table below explains possible error codes and indicates possible corrective actions:

Table 3-4

Error code	Meaning	Remedy
DW#16#000000CA	Watchdog Timer tripped	
DW#16#00048xxx	Input parameter 4 is false. 8xxx displays the precise error message of BLKMOV system block.	The SYS_MSG input variable must be of the format STRING and not longer than 230 characters.
DW#16#00040017	FC17 (INSERT) aborted due to an error.	
DW#16#00040004	FC4 (DELETE) aborted due to an error.	
When using a CP only.		
DW#16#00058xxxx	AG_SEND signals an error. 8xxx displays the precise error message of the AG_SEND communication block (FC5).	Check in NetPro if the configured connection has been duly created and established.
When using a PROFINET CPU only.		
DW#16#00518xxxx	RDSSYST signals an error. 8xxx displays the precise error message of the RDSSYST block (SFC 51).	
DW#16#00658xxx	T_CON signals an error. 8xxx displays the precise error message of the T_CON communication block (FB65).	Check if your syslog server's IP address is correct.
DW#16#00678xxx	T_USEND signals an error. 8xxx displays the precise error message of the T_USEND communication block (FB67).	Check if your syslog server's IP address is correct.
DW#16#00668xxx	T_DISCON signals an error. 8xxx displays the precise error message of the T_DISCON communication block (FB66).	

Note If the error codes DW#16#00658xxx, DW#16#00668xxx or DW#16#00678xxx are displayed you have to upload to the CPU again the whole STEP 7 project after troubleshooting.

4 How to work with the library

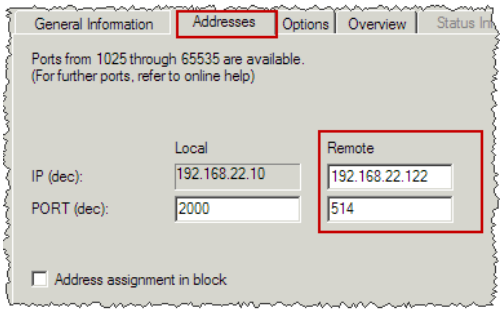
What will you learn here?

This chapter includes directions on how to integrate the SYSLOG library into you STEP 7 project and on how to use the library blocks.

4.1 Preparation

Before you can make use of the library functions the actions below must be accomplished:

Table 4-1

No.	Action	Note
1	Create a STEP 7 project with your hardware configuration.	
2	**Note:** This action is only necessary if you use a communication processor (CP) for data transfer. Configure an unspecified UDP connection. Enter as remote IP address the network address of the PC used as syslog server. 514 must be entered as remote port address.	
3	Install the syslog server.	
4	Install the SQL database (if required)	
5	Network the S7 control system with the syslog server.	

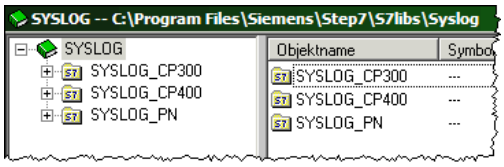
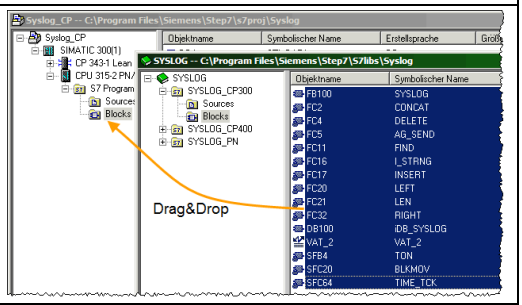
Note The PC with the syslog server software and the control system must be in the same IP network.

4.2 Integrating the library blocks into STEP 7

The actions below define how to integrate the SYSLOG library into your STEP 7 project. Subsequently you can make use of the blocks of the SYSLOG library.

Note The actions below require that a STEP 7 project has already been created.

Table 4-2

No.	Action	Note
1	Unzip the assigned SYSLOG library via **File -> Retrieve** and select a target directory.	
2	The library opens. There are three S7 program folders in the library: • SYSLOG_PN for PROFINET CPU • SYSLOG_CP300 for S7-300CP • SYSLOG_CP400 for S7-400CPs	
3	Open one program folder and copy from **Blocks** all blocks to your blocks container.	
4	Assign error alarms and cyclic interrupts where necessary.	

4.3 Integrating the blocks into the program

Using a call as an example, the following instruction shows how to integrate into your project and assign parameters to the SYSLOG block (FB100).

Assigning parameters to the block

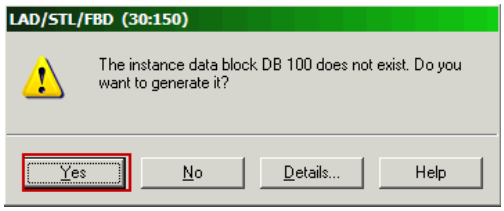
The SYSLOG FB 100 provides input and output parameters which have to be assigned:

Table 4-3

Type	Variable	Data type	Assignment
Input	PRI_SEV	INT	Enter here the severity of the syslog message according to Figure 5-4 in chapter „Message header“. Values between 0 and 7 are accepted.
	PRI_FAC	INT	Enter here the provenience of the syslog message according to Figure 5-4 in chapter „Message header“. Values between 0 and 23 are accepted.
	CONN_ID	BYTE	Defines the ConnectionID for the communication
	LOC_PORT	WORD	Local port for the UDP connection (only available when using the integrated PN interface).
	IP_SERVER	DWORD	If you use a PROFINET CPU, enter here the syslog server's IP address in the format DW#16#00000000. eg. 192.168.0.100 is DW#16#C0A80064.
	CP_ADR	INT	If you use a CP as Ethernet access, enter here the CP's start address from the hardware configuration.
	SYS_MSG	ANY	This variable refers to the syslog message text. The text can have 230 characters maximum.
	START	BOOL	This bit triggers the transfer of the syslog message. The program reacts on a positive edge.
Output	retVal	DWORD	A status message is displayed here if an error occurs.
	error	BOOL	This bit is set to 1 if an error has occurred.

The block call

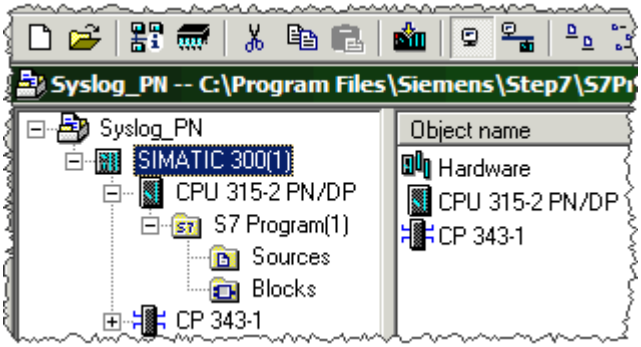
Table 4-4

No.	Action	Note
1		
2	Configure the input and output parameters of FB100.	
3	Save the OB1.	

Download the blocks into the S7 CPU

To ensure that all required blocks are downloaded to the CPU select the SIMATIC 300 station in STEP 7 and download it to your CPU.

Figure 4-1



4.4 Using the library's functionality

The START input parameter of FB100 triggers the transfer of the syslog message.

The record of a positive edge in the SYSLOG FB100 causes the user program to start the state machine and initiate the generation and transfer of the syslog message.

All incoming messages at the syslog server are displayed in a display window.

Figure 4-2

!	Date	Time	Priority	Hostname	Message
	07-21-2011	09:24:44	Auth.Alert	192.168.22.10	Unauthorizised user
	07-21-2011	09:21:47	System0.Info	192.168.22.10	Warmstart of the CPU
	07-21-2011	09:19:12	Local0.Critical	192.168.22.10	Temperature too high

Appendix

5 Syslog protocol

5.1 Description

Syslog as logger to transfer messages within an IP network meanwhile has established as standard (RFC 5424) in the data logging.

Lots of applications are nowadays available which are able to generate syslog records.

Syslog's great advantage is its clear structure and the use in distributed systems. Syslog records may in principle be sent via the network from different computers to one centralized computer and collected there.

Generating a syslog record is very simple: One UDP package is sent to port 514 of a computer where syslog server is running. The UDP package contents must not exceed 1024 characters, must be defined in the US7-ASCII character set and should have the appropriate format.

If required the formatting may transfer the following information to the server:

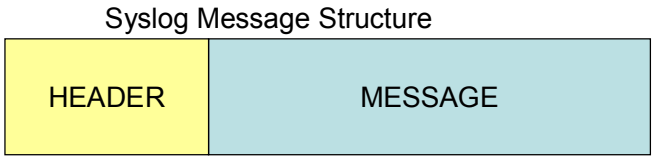
- Priority and type of the package
- Time of its generation
- Name of the source computer.
- Diverse identifiers

Packages with inappropriate formatting are also accepted. The complete contents, though, will be interpreted as message text in this case. Default values will replace parameters which could not be identified (like the time of its generation).

5.2 Syslog protocol structure

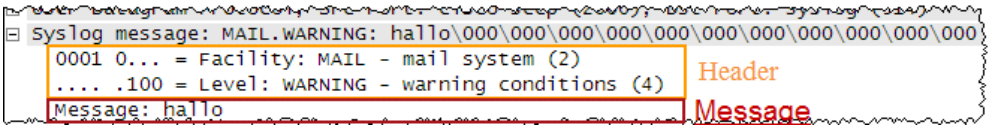
The syslog protocol has a very simple structure which can be divided into two essential blocks: the header and the proper message.

Figure 5-1



The chart below shows a syslog message recorded via the "Wireshark" analysis tool:

Figure 5-2



5.3 Message header

Description

The header manages information regarding:

- the type of message
- the time
- a version ID
- the host name.

Note

Except for the "Type of message" the syslog client cannot make further modifications at the header.

The rest of parameters will all be filled with values by the syslog server.

Formatting

The characters must be of ASCII (7 bit) format in an 8-bit-field.

The section below shows the ASCII character set.

Figure 5-3

Scan-code	ASCII hex	dez	Zeichen	Scan-code	ASCII hex	dez	Zch.	Scan-code	ASCII hex	dez	Zch.	Scan-code	ASCII hex	dez	Zch.
	00	0	NUL ^@		20	32	SP		40	64	@	0D	60	96	`
	01	1	SOH ^A	02	21	33	!	1E	41	65	A	1E	61	97	a
	02	2	STX ^B	03	22	34	"	30	42	66	B	30	62	98	b
	03	3	ETX ^C	29	23	35	#	2E	43	67	C	2E	63	99	c
	04	4	EOT ^D	05	24	36	$	20	44	68	D	20	64	100	d
	05	5	ENQ ^E	06	25	37	%	12	45	69	E	12	65	101	e
	06	6	ACK ^F	07	26	38	&	21	46	70	F	21	66	102	f
	07	7	BEL ^G	0D	27	39	'	22	47	71	G	22	67	103	g
0E	08	8	BS ^H	09	28	40	(	23	48	72	H	23	68	104	h
0F	09	9	TAB ^I	0A	29	41	)	17	49	73	I	17	69	105	i
	0A	10	LF ^J	1B	2A	42	*	24	4A	74	J	24	6A	106	j
	0B	11	VT ^K	1B	2B	43	+	25	4B	75	K	25	6B	107	k
	0C	12	FF ^L	33	2C	44	,	26	4C	76	L	26	6C	108	l
1C	0D	13	CR ^M	35	2D	45	-	32	4D	77	M	32	6D	109	m
	0E	14	SO ^N	34	2E	46	.	31	4E	78	N	31	6E	110	n
	0F	15	SI ^O	08	2F	47	/	18	4F	79	O	18	6F	111	o
	10	16	DLE ^P	0B	30	48	0	19	50	80	P	19	70	112	p
	11	17	DC1 ^Q	02	31	49	1	10	51	81	Q	10	71	113	q
	12	18	DC2 ^R	03	32	50	2	13	52	82	R	13	72	114	r
	13	19	DC3 ^S	04	33	51	3	1F	53	83	S	1F	73	115	s
	14	20	DC4 ^T	05	34	52	4	14	54	84	T	14	74	116	t
	15	21	NAK ^U	06	35	53	5	16	55	85	U	16	75	117	u
	16	22	SYN ^V	07	36	54	6	2F	56	86	V	2F	76	118	v
	17	23	ETB ^W	08	37	55	7	11	57	87	W	11	77	119	w
	18	24	CAN ^X	09	38	56	8	2D	58	88	X	2D	78	120	x
	19	25	EM ^Y	0A	39	57	9	2C	59	89	Y	2C	79	121	y
	1A	26	SUB ^Z	34	3A	58	:	15	5A	90	Z	15	7A	122	z
01	1B	27	Esc ^[	33	3B	59	;		5B	91	[		7B	123	{
	1C	28	FS ^\	2B	3C	60	<		5C	92	\		7C	124	|
	1D	29	GS ^]	0B	3D	61	=		5D	93	]		7D	125	}
	1E	30	RS ^^	2B	3E	62	>	29	5E	94	^		7E	126	~
	1F	31	US ^_	0C	3F	63	?	35	5F	95	_	53	7F	127	DEL

Structure

The syslog protocol dictates a fixed order and structure of the header parameters. If these rules are neglected the syslog server cannot interpret the information accordingly.

The following structure applies:

PRI VERSION SP TIMESTAMP SP HOSTNAME SP APP-NAME SP PROCID SP MSGID

It is not imperative that all elements are included in a syslog message. Default values are assigned to such parameters which cannot be identified.

Note All elements and parameters must be entered into the header in ASCII format (7 bits).

The parameters have the following meaning:

Table 5-1

Parameters	Meaning
PRI	The PRI range must be limited by the "< (%d60)" and "> (%d62)" characters and is three to five characters long. The priority of the syslog message - subdivided into Severity and Facility fields - is within PRI.
VERSION	The version ID may comprise up to two bytes but must include characters between 1 and 9 (%d49-57) only. The version number of the syslog specification may be indicated in this field.
HOSTNAME	HOSTNAME references the source computer with its name and IP address. It may be between 1 and 255 characters long and include all characters between %d33-126. The "-" character is output if the source computer is unidentified.
APP-NAME	APP-NAME includes the application name. It can be from 1 to 48 characters long. All characters comprised between %d33-126 are accepted. If the application name is unidentified, "-" is output.
PROCID	PROCID includes the process ID as information. It can be from 1 to 128 characters long. All characters comprised between %d33-126 are accepted. If the ID is unidentified, "-" is output.
MSGID	This parameter identifies the message and provides a length between 1 and 32 characters. All characters comprised between %d33-126 are accepted. If the ID is unidentified, "-" is output.
SP	Corresponds to ASCII code %d32.
TIMESTAMP	This range comprises the time stamp and has its own structure.

Note Refer to RFC 5424 for further information on the parameters, please.

http://tools.ietf.org/html/rfc5424

Encoding for the PRI range

PRI stands for priority and defines the provenience (Facility field) and the severity (Severity field) of the message. This parameter is the only one which can be modified via syslog client.

The Facility field provides up to 5 bits the numerical value of which indicates the service or the facility having generated the syslog message.

An extract of the RFC 5424 shows the possible range of values:

Figure 5-4

Numerical Code	Facility
0	kernel messages
1	user-level messages
2	mail system
3	system daemons
4	security/authorization messages
5	messages generated internally by syslogd
6	line printer subsystem
7	network news subsystem
8	UUCP subsystem
9	clock daemon
10	security/authorization messages
11	FTP daemon
12	NTP subsystem
13	log audit
14	log alert
15	clock daemon (note 2)
16	local use 0 (local0)
17	local use 1 (local1)
18	local use 2 (local2)
19	local use 3 (local3)
20	local use 4 (local4)
21	local use 5 (local5)
22	local use 6 (local6)
23	local use 7 (local7)

The Severity field provides three bits the numerical value of which defines the severity of the syslog message.

An extract of the RFC 5424 shows the possible range of values:

Figure 5-5

Numerical Code	Severity
0	Emergency: system is unusable
1	Alert: action must be taken immediately
2	Critical: critical conditions
3	Error: error conditions
4	Warning: warning conditions
5	Notice: normal but significant condition
6	Informational: informational messages
7	Debug-level messages

The value to be entered between the characters "<[value of priority]>" (ASCII-encoded) is calculated as follows:

Priority value = Facility value * 8 + Severity.

Example:

A "local use 4" message (facility = 20) with the severity of "notice" (severity = 5) has a priority value of 20*8 + 5 = 165.
This result is to be placed between angle brackets as ASCII character. In this case parameter PRI in the header is five bytes long and includes "<165>" as value or "%d60 %d49 %d54 %d53 %d62" expressed as a decimal.

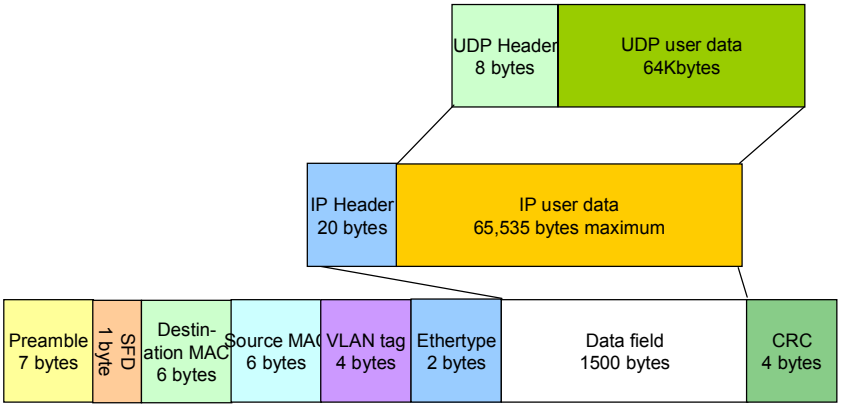
5.4 How the message is transferred

Syslog uses UDP/IP and Ethernet as transmission protocol.
UDP is a protocol without connection and thus, not reliable. This means that transmission cannot be guaranteed 100 percent successful.

For their transfer the syslog messages are packed into the user data range of the UDP frame. The syslog messages could theoretically occupy the complete capacity of the UDP user data range (64kByte). The size of a syslog message however, is limited to the maximum size of the Ethernet user data range because the UDP frame itself will be packed into the IP frame's user data range, which in turn is packed into the Ethernet's user data range.

The Ethernet data field merely includes 1500 bytes. Owing to the header's overhead (IP (20 bytes), UDP (8 bytes) and the syslog message) the syslog message text may have a maximum length of 1024 bytes.

Figure 5-6



6 SQL database connection

What will you learn here?

This chapter provides help in handling the described SYSLOG library.

It explains how to proceed to carry out a SQL database connection between syslog server and SQL database.

Note

The directions for configuration and the figures are based on the software below:

- KIWI Syslog Server Version 9.2
- MySQL-Server V5.5

6.1 Precondition

Installing the SQL database

To connect the syslog server to a SQL database a SQL database needs to be installed, apart from the syslog server software.

The installation of a SQL Manager Tool helps configure the database. The tool offers a user-friendly interface which spares the user the configuration of the SQL database via commands and SQL commands.

Installing the OLE DB driver

The ODBC connection is used to connect the syslog server to the SQL database.

As the MySQL database is coming without OLE DB driver it must be installed additionally.

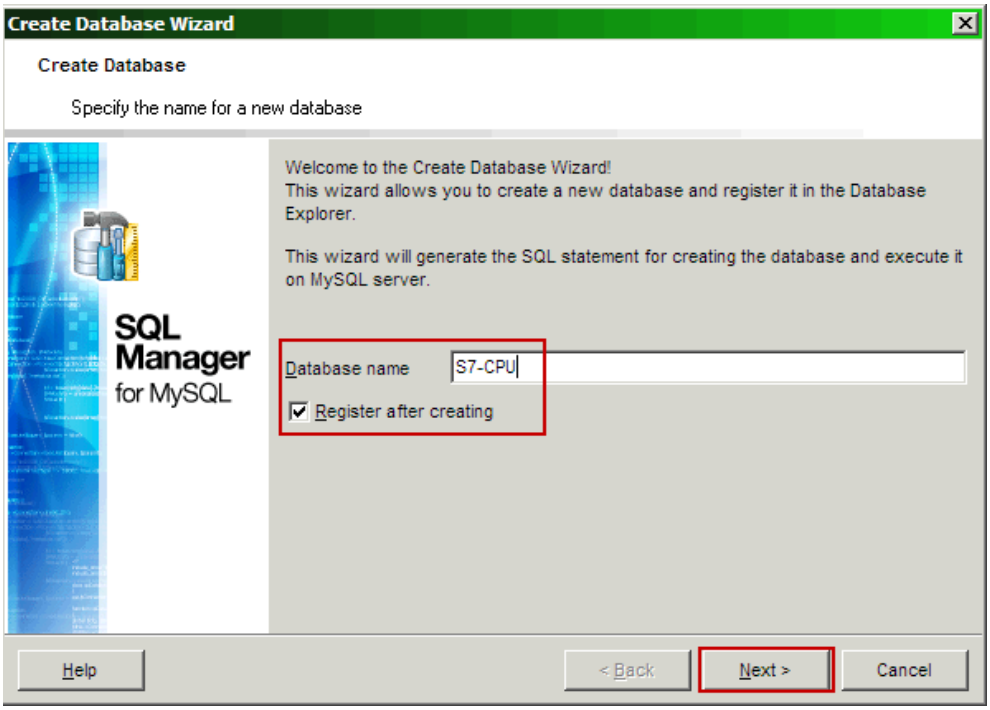
The OLE DB driver can be downloaded from the MySQL website.

The MYSQL ODBC 5.1 driver is used below.

6.2 Creating a new database

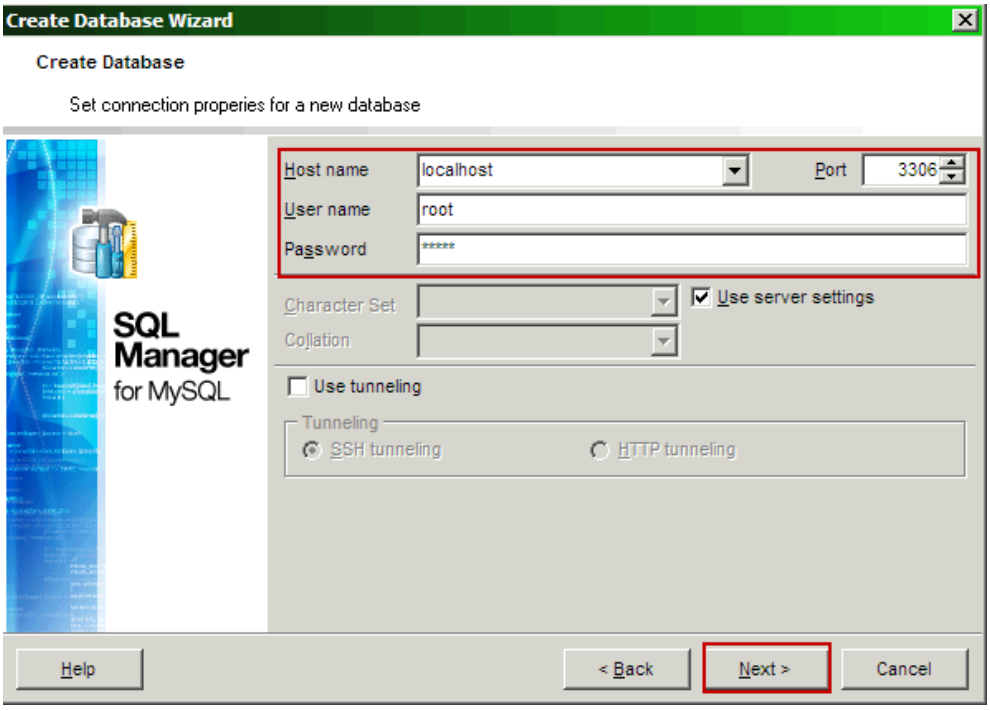
Open the SQL configuration manager and create a new database.

Figure 6-1



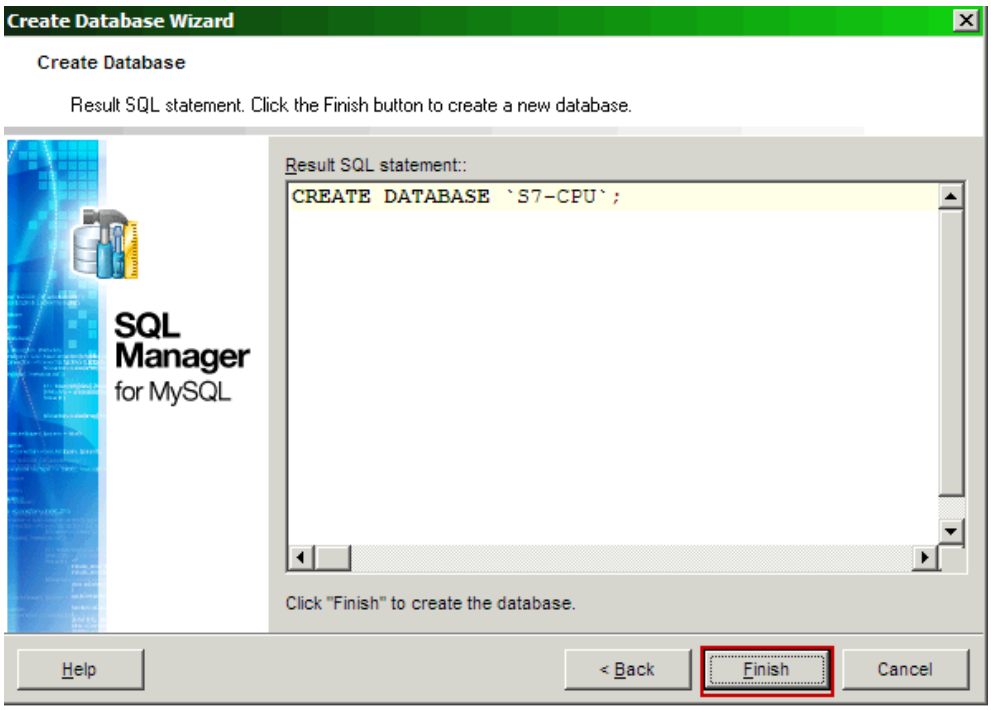
Define the connection properties.

Figure 6-2



Finalize the configuration of the new database.

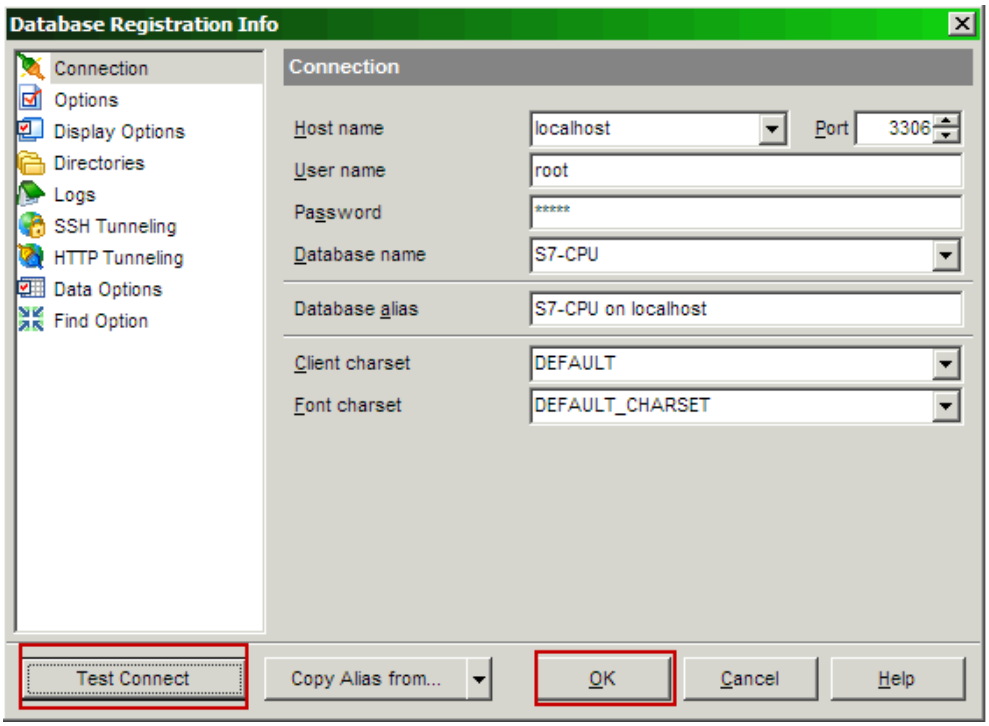
Figure 6-3



A new dialog window informs you on the parameters you have just configured. An option to test the connection to the new database is available.

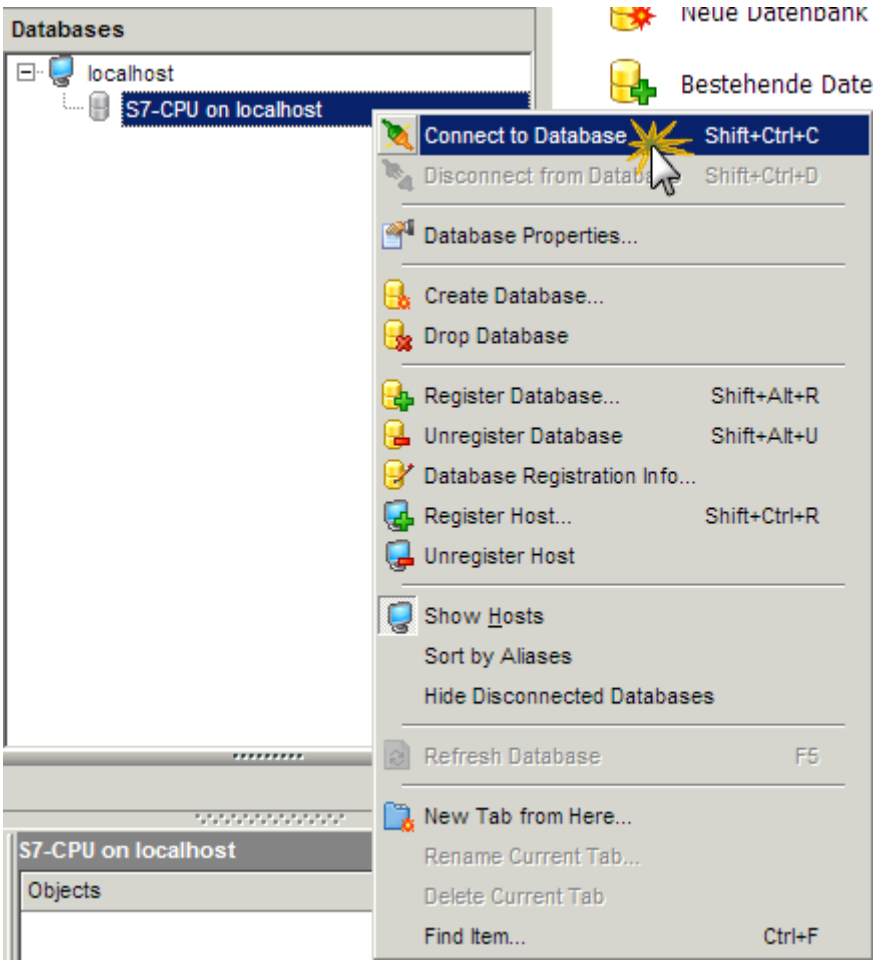
Finalize the configuration if the test was successful.

Figure 6-4



Connect to the new database.

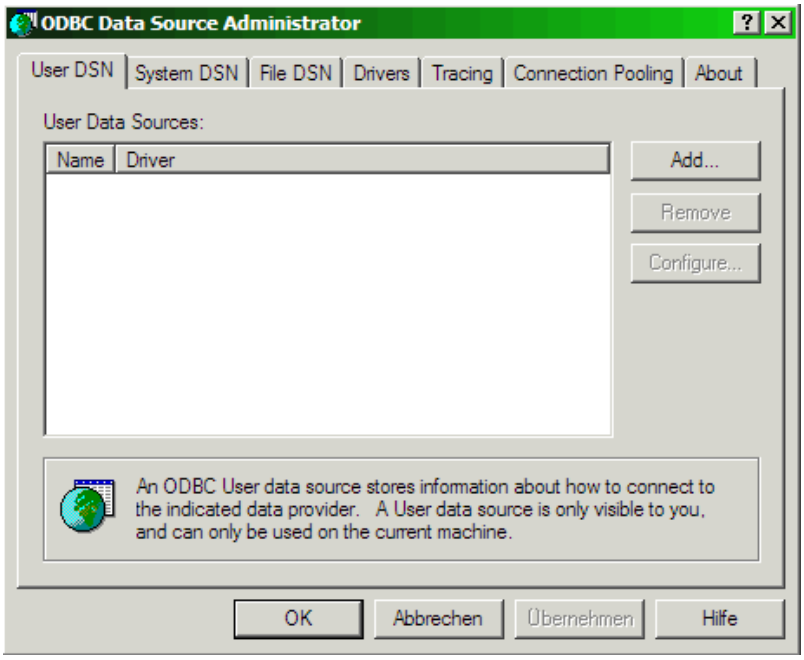
Figure 6-5



6.3 Configuring the ODBC data source

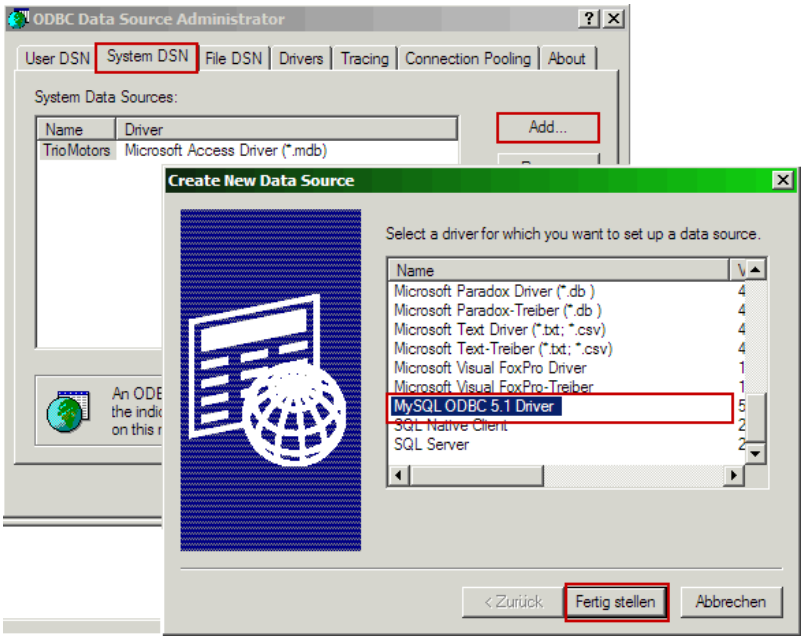
Open the ODBC data source via Control Panel > Administrative Tools.

Figure 6-6



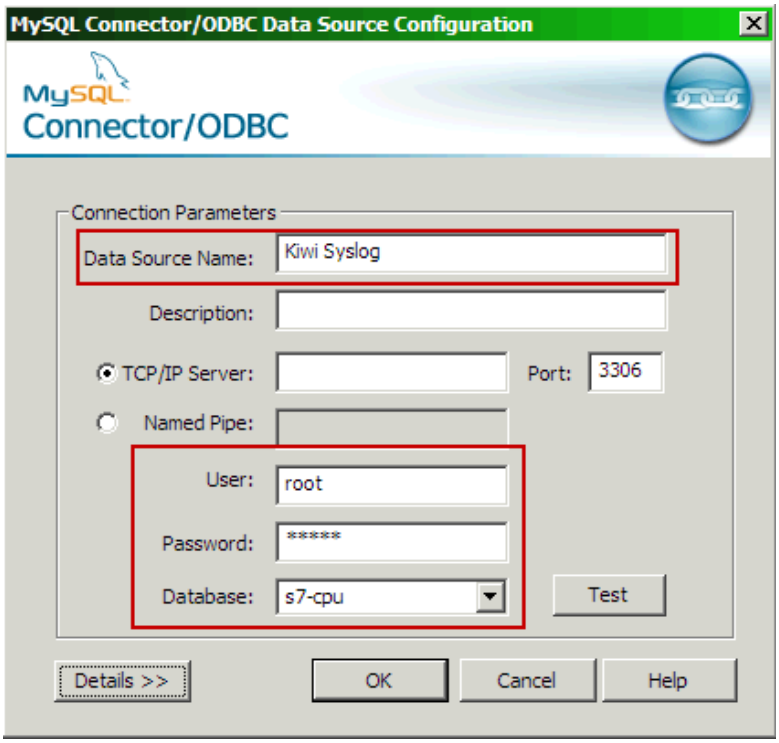
Create a new data structure via **Add...** in the **System DSN** tab. Select the MySQL ODBC driver you installed before for this purpose.

Figure 6-7



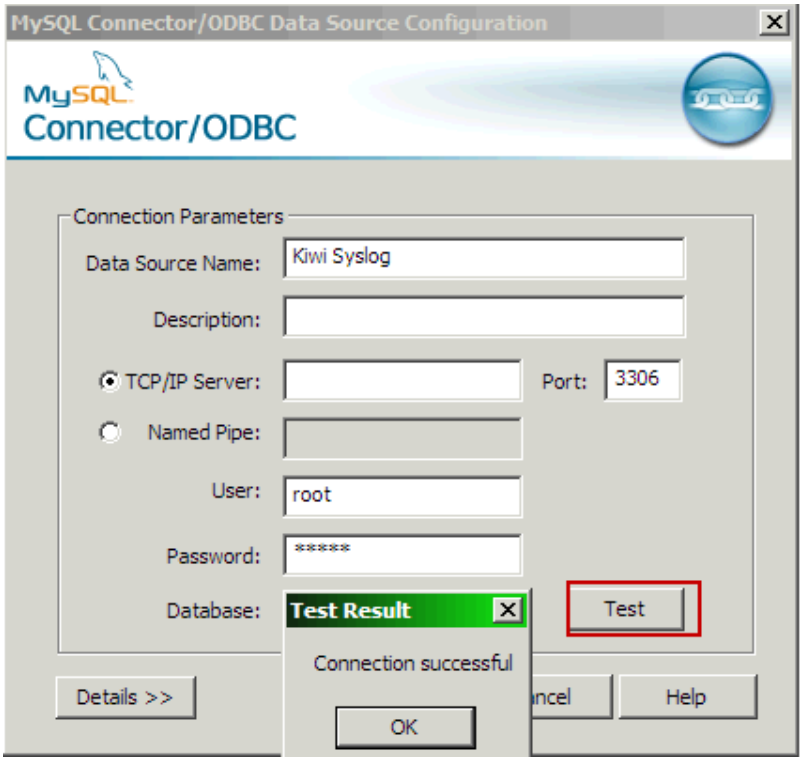
Assign a name to the new data structure. Define the user, the related password and the name of the database you have created.

Figure 6-8



Press the Test button to check if the settings are correct and the server is connected to the database.

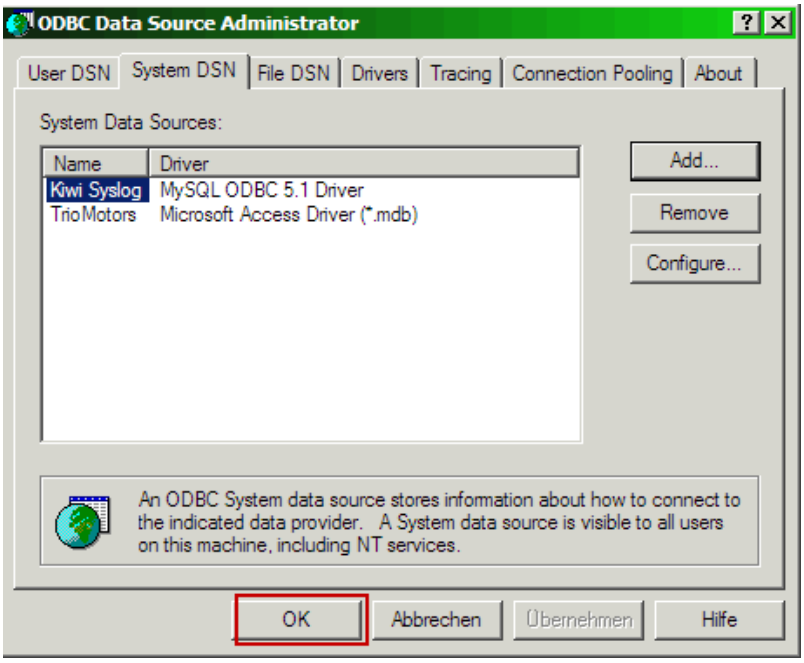
Figure 6-9



Complete the configuration with "OK".

Select the data structure you have created before and close the ODBC data source.

Figure 6-10

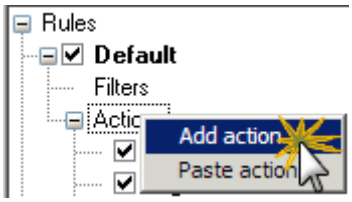


6.4 Settings in the syslog server

Creating a new action

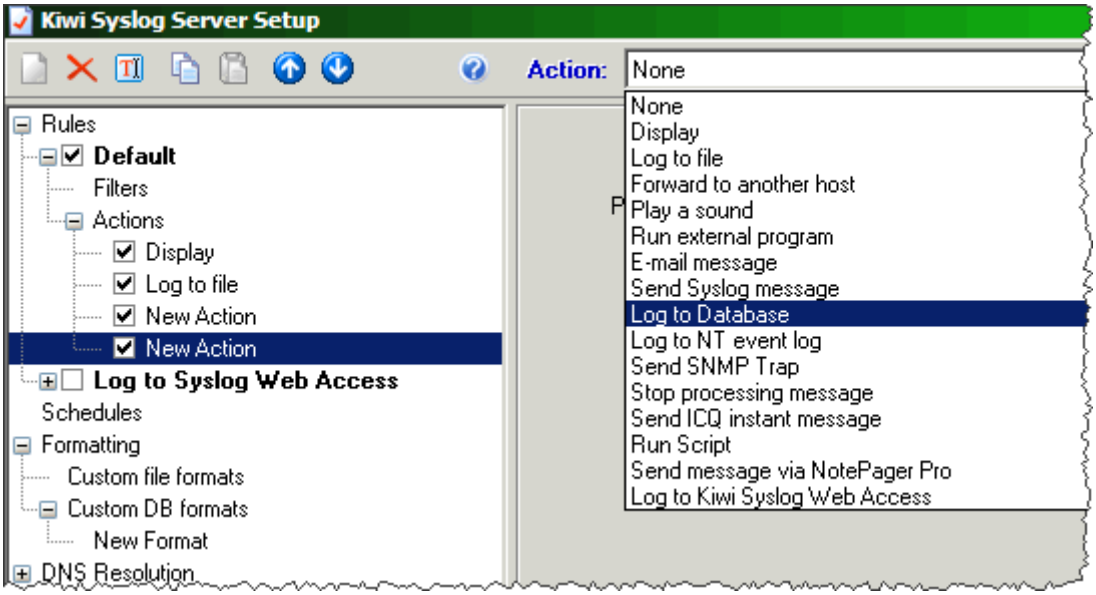
In the syslog server, open the setup window via File > Setup. Create a new action in the menu item Rules > Default.

Figure 6-11



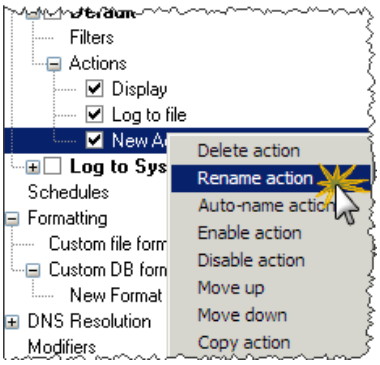
Select the Log to Database option as action.

Figure 6-12



The default name may be changed to reveal the function of the new action.

Figure 6-13



Defining the data connection

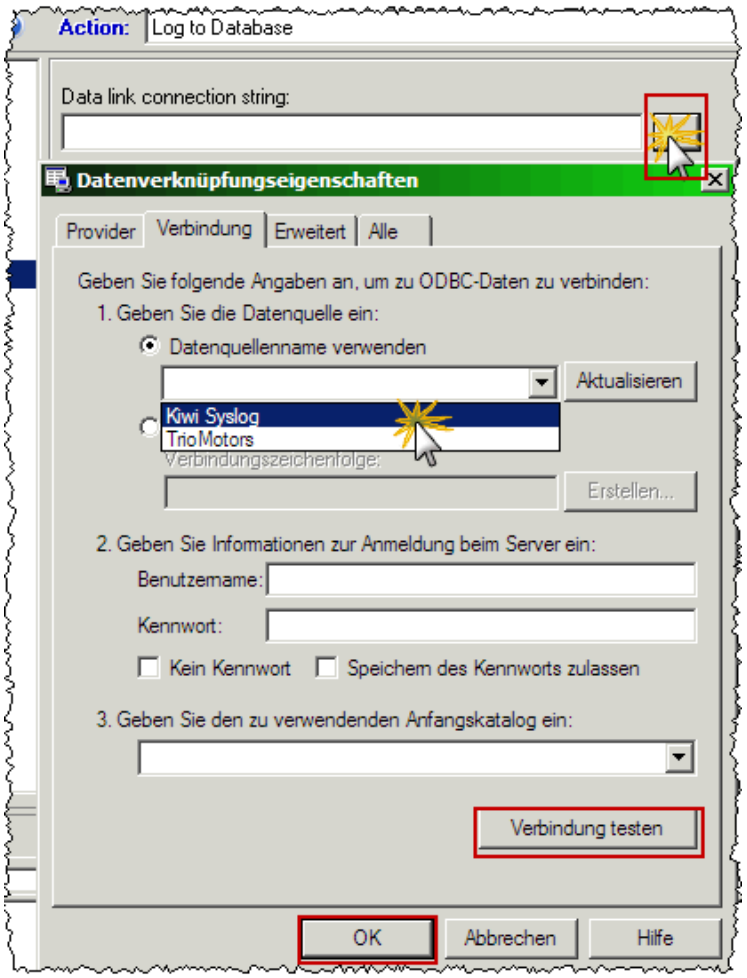
The next step is to define the data connection to the SQL database.

Select from the pick list the data source created in section 6.3 (Configuring the ODBC data source

).

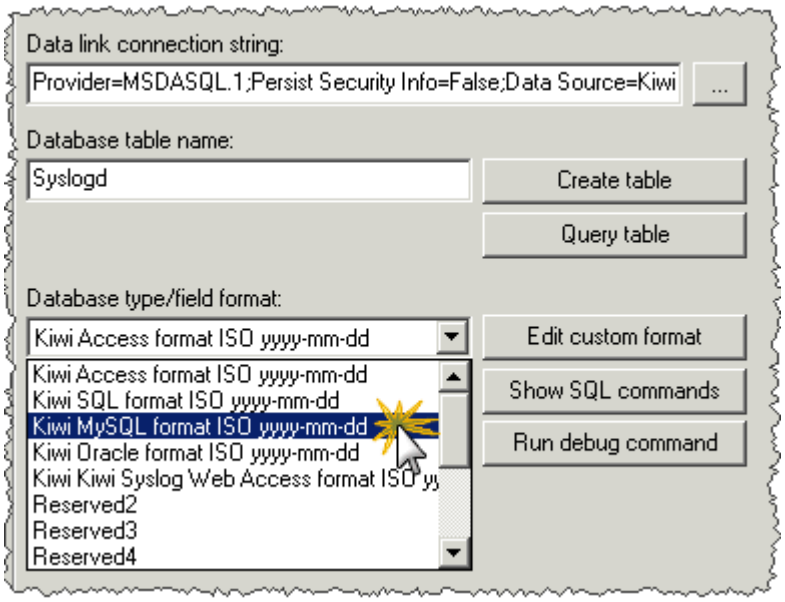
You can test the connection here, too.

Figure 6-14



Select the MySQL database format.

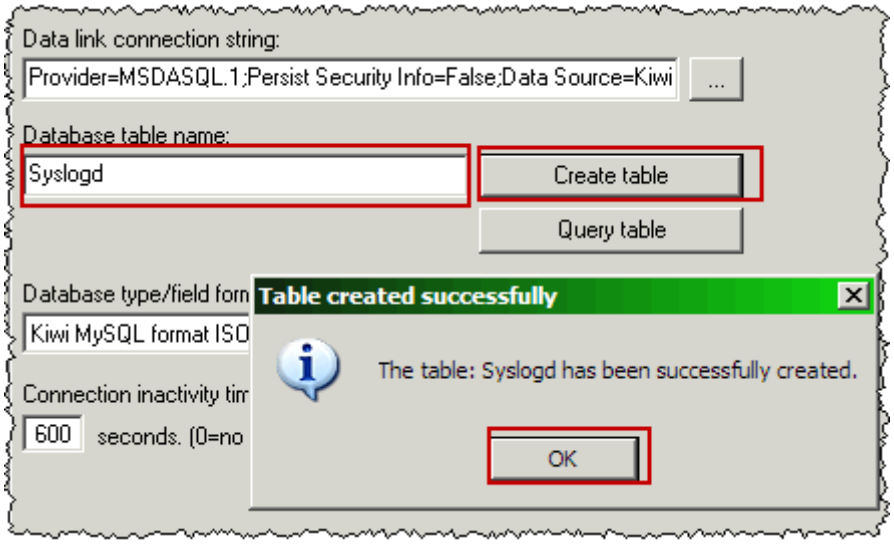
Figure 6-15



Creating a new database table

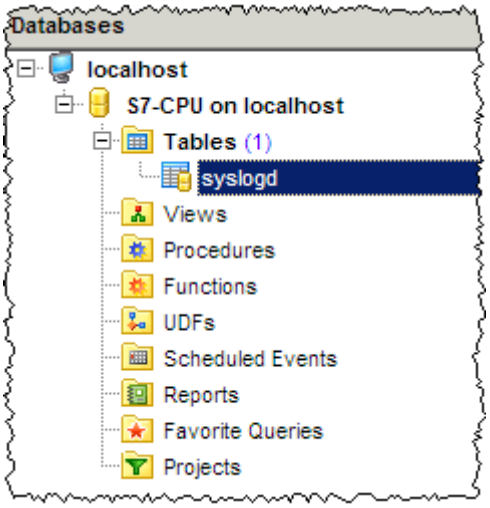
Change the default name of the data base table - if you wish so - and create a new table via the corresponding button. A dialog window informs you on the generation status.

Figure 6-16



The SQL Manager Tool displays the newly created table.

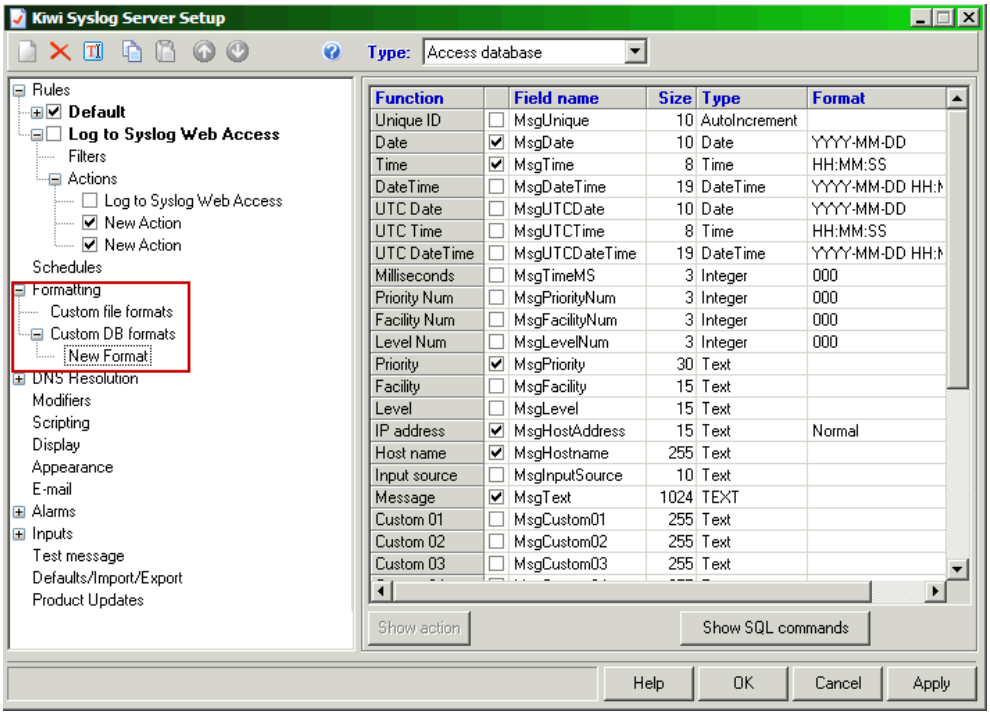
Figure 6-17



Adapting the table entries

You can adapt to your requirements the database table columns and add or remove columns.

Figure 6-18



Result

Apart from the visualization on the screen all syslog messages will also be stored to the SQL database owing to this configuration.

Note Messages transferred from the syslog server to the SQL database must not have characters of the type "%d0" (NUL) in the message text.

7 Links & References

7.1 References

This list does not claim to be exhaustive and reflects a selection of related references only.

Table 7-1

	Topic	Title
/1/	STEP7	Automatisieren mit STEP7 in AWL und SCL (Automating with STEP7 in STL and SCL) Hans Berger Publicis Corporate Publishing ISBN 3-89578-113-4
/2/		

7.2 Internet links

The following list does not claim to be exhaustive and reflects a selection of related sources only.

Table 7-2

	Topic	Title
\1\	Reference to the document	http://support.automation.siemens.com/WW/view/en/51929235
\2\	Siemens I IA/DT Customer Support	http://support.automation.siemens.com

8 History

Table 8-1

Version	Date	Revisions
V1.0	15.08.2011	First issue
V1.1	22.06.2012	Additional Inputparameters for Fb100 (Local port, connectionID) The format of the IP-address is know DWORD.