



Automatic Generation of Qualification Documentation at the Example of RTEMS QDPs

Frank Kühndel, Sebastian Huber and Matthias Göbel

embedded brains GmbH & Co. KG
Puchheim, Germany



Overview



Introduction

- RTEMS OS space qualified to category B
- Automation for Cheaper and Frequent QDP Updates
- Manual Work Is Still Needed

Automation

- RTEMS QDPs build by an automated process
- Documents generated with data from the build process

Examples

- Example spec item use
- Generated C header files add hyperlinks
- Doxygen tag file helps to create traceability matrices

Conclusion



RTEMS OS Space Qualified to Cat. B



...and it works!

- RTEMS = Real Time OS, Open Source Software
- Qualified Data Package (QDP)
 - Pre-Qualified to Cat. B (ECSS standard; checked by ESA)
 - For Gaisler SPARC processors for aerospace (radiation-hard):
GR712RC (LEON3 dual core) and GR740 (LEON4 quad core)
 - Download at <https://rtems-qual.io.esa.int/>
- Currently qualification update of multi and single core QDPs



Automation for Economic and Frequent QDP Updates



...and it works!

Open source RTEMS code is **continuously** developed

- Many extensions
- Many patches
- Many updates

→ **very dynamic**



Qualification is labor-intensive

→ **very static**

Automatically Generated (in parts)

- ICD – Interface Control Document
- SCF – Software Configuration File
- SDD – Software Design Document
- SPAMR – Software Product Assurance Record
- SRELD – Software Release Document
- SRS – Software Requirements Specification
- SUITP – Software Unit and Integration Test Plan
- SVR – Software Verification Record includes
 - SUITR – Software Unit and Integration Test Report
 - SVaIR – Software Validation Report
- SVS – Software Validation Specification
- SUM – Software User Manual
- RTEMS Software Engineering
- RTEMS CPU Architecture Supplement
- RTEMS Classic API Guide

Manual Work Is Still Needed

Manually Written

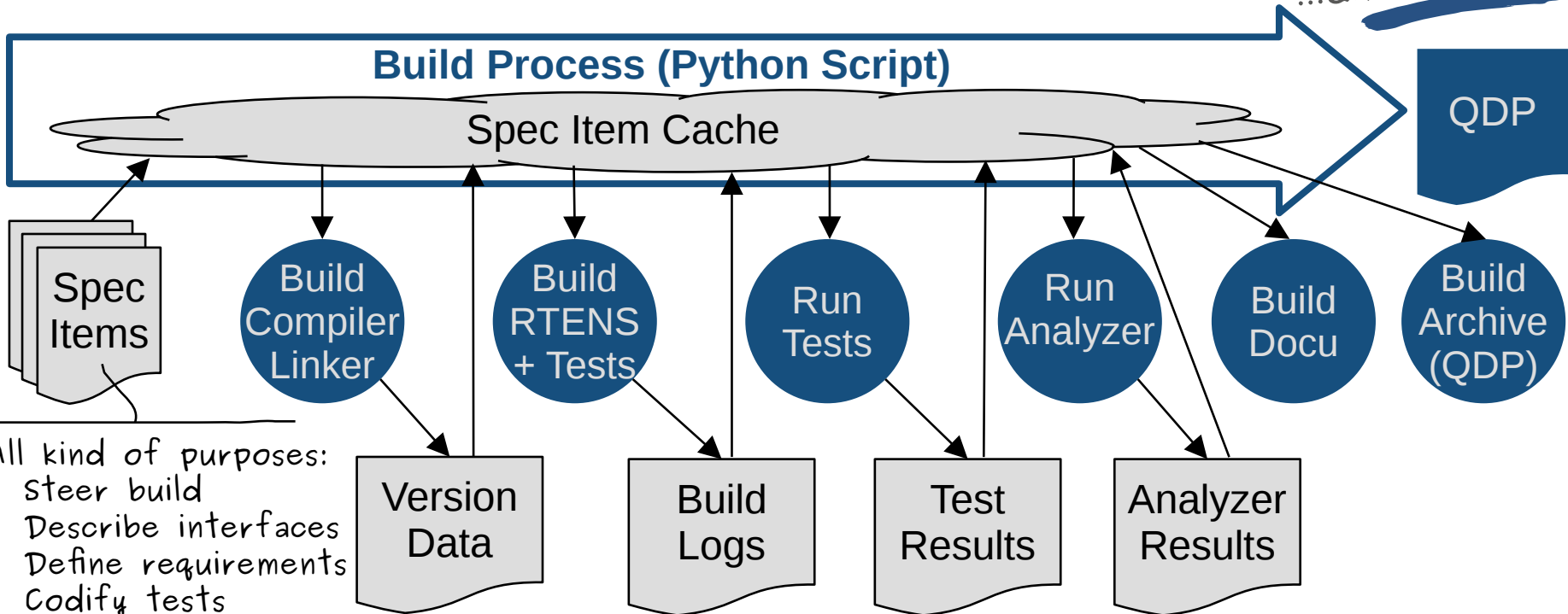
- Formal Verification
- Software Reuse File
- Software Review Plan
- Software Configuration Management Plan
- Software Development Plan
- Software Product Assurance Plan
- Tool Chain Software
- User Manual
- Technical Notes



RTEMS QDPs Build by an Automated Process



...and it works!



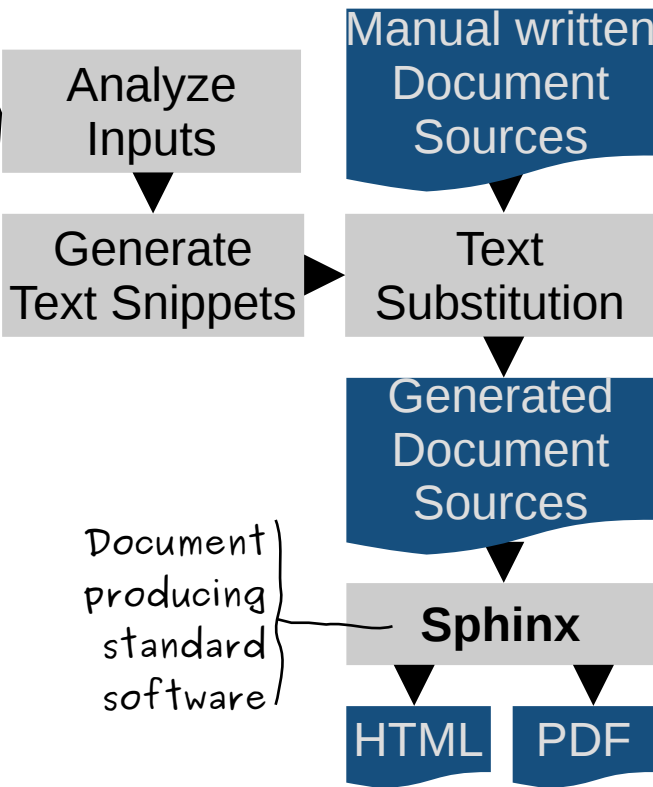
- All kind of purposes:
- Steer build
 - Describe interfaces
 - Define requirements
 - Codify tests
 - Define Glossary
 - ...



Documents Generated with Data from the Build



- Configuration
- Build Log-Files
- Test Results
- Coverage Results
- Static Analyzer
- ...



Verification of Software Unit Testing
AA

The following table contains the verification of the status of each unit test specification as presented in |SUITP|.

```
#{./unit-verification}
```

Replace with generated table

4.4.2.8 Verification of Software Unit Testing

The following table contains the verification of the status of each unit test specification as presented in Software Unit and Integration Test Plan (SUITP).

Table 7: Unit Test Status

Test specification	Test name	Status
spec:/score/rbtree/unit/rbtree	ScoreRbtreeUnitRbtree	P, P
spec:/score/msgq/unit/msgq	ScoreMsgqUnitMsgq	P, P
spec:/rtems/config/unit/config	RtemsConfigUnitConfig	P, P



Example Spec Item Use (1/4)



...and it works!

rtems/timer/if/create.yml ^{Spec item/YML file}

ICD

```
brief: |  
  Creates a timer.  
description: |  
  This directive creates a timer  
  which resides on the local  
  Node. [...]  
name: rtems_timer_create  
params:  
- description: |  
  is the object name of the  
  timer.  
  name: name  
[...]
```



5.2.2.685. spec:/rtems/timer/if/create

interface-type:

function

name:

rtems_timer_create

description:

This directive creates a timer which resides on the local node.
object name specified in name. The assigned object identifier
used to access the timer with other timer related directives.

function:

```
rtems_status_code rtems_timer_create( rtems_name name, rte
```




Example Spec Item Use (3/4)



...and it works!

rtems/timer/if/create.yml

```
brief: |
  Creates a timer.
description: |
  This directive creates a timer which resides on the local Node. [...]
name: rtems_timer_create
params:
- description: |
  is the object name of the timer.
  name: name
  [...]
```



RTEMS Classic API Guide

10.4.1. rtems_timer_create()

Creates a timer.

CALLING SEQUENCE:

```
rtems_status_code rtems_timer_create( rtems_name name,
```

PARAMETERS:

name

This parameter is the object name of the timer.

id

This parameter is the pointer to an **rtems_id** object. It



Example Spec Item Use (2/4)



...and it works!

rtems/timer/if/create.yml

```
brief: |
  Creates a timer.
description: |
  This directive creates a timer
  which resides on the local
  Node. [...]
name: rtems_timer_create
params:
- description: |
  is the object name of the
  timer.
  name: name
  [...]
```



rtems/timer.h

```
/* Generated from spec:/rtems/timer/if/create */
.
/**.
 * @ingroup RTEMSAPIClassicTimer.
 *
 * @brief Creates a timer..
 *
 * @param name is the object name of the timer.
 *
 * @param[out] id is the pointer to an ::rtems_
 * call is successful, the identifier of the
 * this object..
 *
 * This directive creates a timer which resides
 * has the user-defined object name specified in
 * identifier is returned in ``id``. This ident
 * timer with other timer related directives..
 *
 */
```



Example Spec Item Use (4/4)



...and it works!

rtems/timer/if/create.yml

```
brief: |  
  Creates a timer.  
description: |  
  This directive creates a timer  
  which resides on the local  
  Node. [...]  
name: rtems_timer_create  
params:  
- description: |  
  is the object name of the  
  timer.  
  name: name  
  [...]
```



A number of „smaller“ usages
(e.g. the function name in
requirements)

Traceability matrices
(I will show ...)

Hyperlinks between documents
(I will show ...)

**Avoiding repetition is not the
only usage**

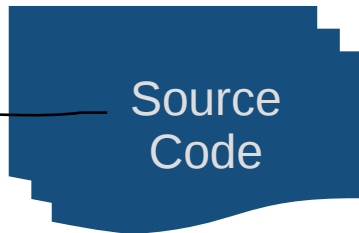


Generated C Header Files Add Hyperlinks (1/3)



...and it works!

Includes the generated
rtems/timer.h
⇒ SDD contains the text
from the spec item, too



Doxygen

Standard software to
create
documentation
from source
code



SDD

```
◆ rtems_timer_create()
rtems_status_code rtems_timer_create ( rtems_name name,
Home rtems_id * id
)

Creates a timer.

Parameters
name is the object name of the timer.
[out] id is the pointer to an rtems_id object. When the directi
object.

This directive creates a timer which resides on the local node. The timer
identifier is returned in id. This identifier is used to access the timer with

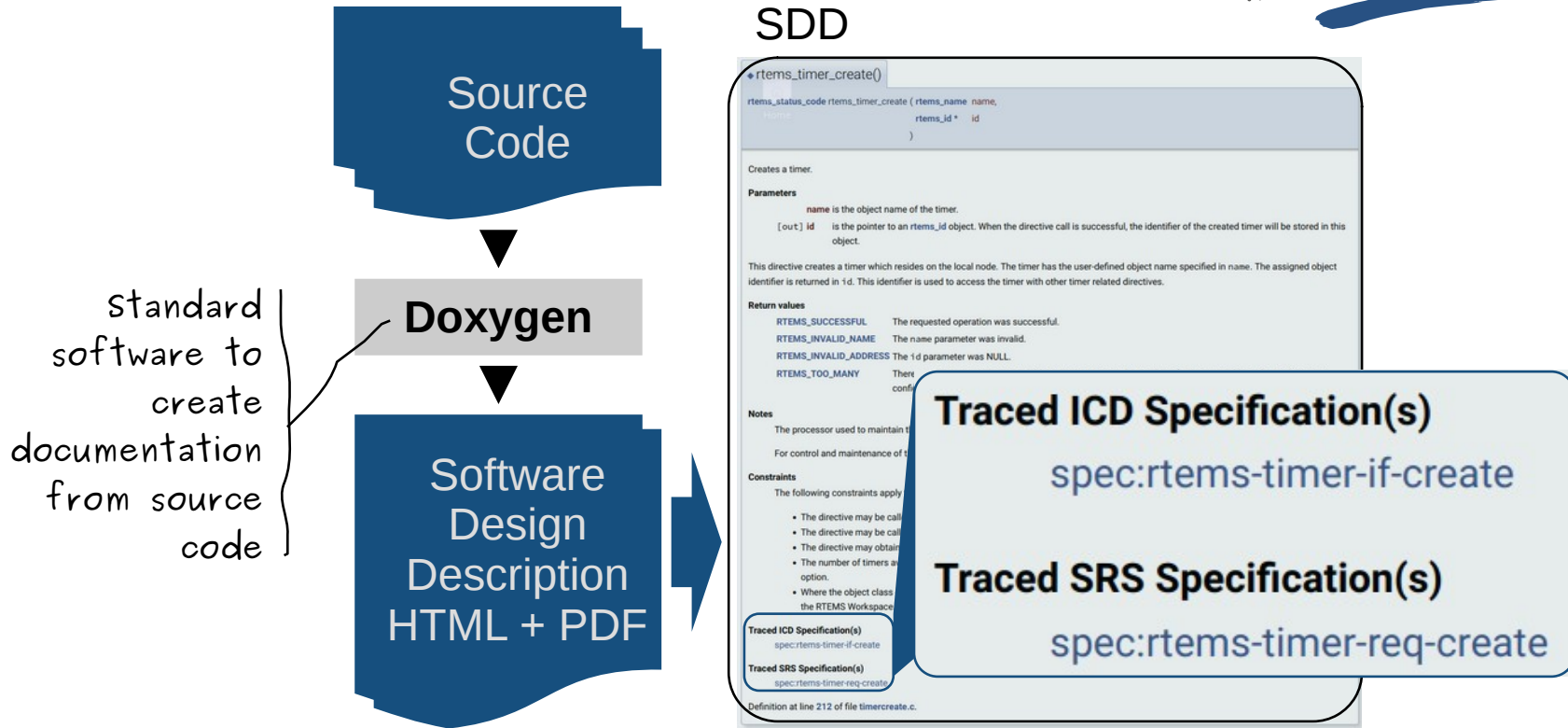
Return values
```



Generated C Header Files Add Hyperlinks (2/3)



...and it works!





Generated C Header Files Add Hyperlinks (3/3)



...and it works!

Generated files added as input

LinksToSrs.h
LinksToIcd.h

Source Code

Doxygen

Software Design Description
HTML + PDF

LinksToIcd.h

```
/**
 * @fn rtems_status_code.
 *      rtems_timer_create(
 *          rtems_name name,
 *          rtems_id *id).
 *
 *
 * @par Traced ICD Specification(s).
 * @parblock.
 * <a href='requirements_and_design.htm
 * spec-rtems-timer-if-create'>.
 * spec:rtems-timer-if-create</a>.
 * @endparblock.
 */
```

SDD

```

rtems_timer_create()
rtems_status_code rtems_timer_create ( rtems_name name,
                                       rtems_id * id
                                       )

Creates a timer.

Parameters
name is the object name of the timer.
[out] id is the pointer to an rtems_id object. When the directive call is successful, the identifier of the created timer will be stored in this object.

This directive creates a timer which resides on the local node. The timer has the user-defined object name specified in name. The assigned object identifier is returned in id. This identifier is used to access the timer with other timer related directives.

Return values
RTEMS_SUCCESSFUL The requested operation was successful.
RTEMS_INVALID_NAME The name parameter was invalid.
RTEMS_INVALID_ADDRESS The id parameter was NULL.
RTEMS_TOO_MANY There are too many timers configured.

Notes
The processor used to maintain the timer.
For control and maintenance of the timer.

Constraints
The following constraints apply
• The directive may be called multiple times.
• The directive may be called from any task.
• The directive may obtain the object class of the timer.
• The number of timers a node can have is limited by the configuration option.
• Where the object class is RTEMS_TIMER, the timer is created in the RTEMS Workspace.

Traced ICD Specification(s)
spec:rtems-timer-if-create

Traced SRS Specification(s)
spec:rtems-timer-req-create

Definition at line 212 of file timercreate.c.

```

Traced ICD Specification(s)
spec:rtems-timer-if-create
Traced SRS Specification(s)
spec:rtems-timer-req-create



Doxygen Tag File Helps to Create Traceability Matrices (1/3)



...and it works!

Source Code

Doxygen

Software Design Description
HTML + PDF

Doxygen tagfile

```
<member kind="function">.  
  <type>rtems_status_code</type>.  
  <name>rtems_timer_create</name>.  
  <anchorfile>group__RTEMSAPIClassicTime</anchorfile>.  
  <anchor>ga57cf664e958961aefdb8ef8440f2</anchor>.  
  <arglist>(rtems_name name, rtems_id *id)</arglist>.  
</member>.
```

tagfile

Tagfile generated by Doxygen

- Lists all source code elements
- Provides hyperlink into SDD



Doxygen Tag File Helps to Create Traceability Matrices (2/3)



...and it works!

Generating the ICD and SRS:
Use tagfile info to create cross references to SDD



ICD

Traced design component:

RTEMSAPIClassicTimer - rtems_timer_create

SRS

Traced design component:

RTEMSAPIClassicTimer - rtems_timer_create

Doxygen tagfile

```
<member kind="function">.
  <type>rtems_status_code</type>.
  <name>rtems_timer_create</name>.
  <anchorfile>group__RTEMSAPIClassicTime
  <anchor>ga57cf664e958961aefdb8ef8440f2
  <arglist>(rtems_name name, rtems_id *id
</member>.
```

tagfile

Tagfile generated by Doxygen

- Lists all source code elements
- Provides hyperlink into SDD



Doxygen Tag File Helps to Create Traceability Matrices (3/3)



...and it works!

⇒ Tagfile helps generate traceability matrices and linking documents

ICD

Traced design component:

RTEMSAPIClassicTimer - rtems_timer_create

SRS

Traced design component:

RTEMSAPIClassicTimer - rtems_timer_create

SDD

```

rtems_timer_create()
rtems_status_code rtems_timer_create ( rtems_name name,
                                       rtems_id * id
                                       )

Creates a timer.

Parameters
  name is the object name of the timer.
  [out] id is the pointer to an rtems_id object. When the directive call is successful, the identifier of the created timer will be stored in this object.

This directive creates a timer which resides on the local node. The timer has the user-defined object name specified in name. The assigned object identifier is returned in id. This identifier is used to access the timer with other timer related directives.

Return values
  RTEMS_SUCCESSFUL The requested operation was successful.
  RTEMS_INVALID_NAME The name parameter was invalid.
  RTEMS_INVALID_ADDRESS The id parameter was NULL.
  RTEMS_TOO_MANY There are too many timers configured.

Notes
  The timer is used to maintain the time of day.
  For control and management of the timer, see the rtems_timer_create() and rtems_timer_delete() directives.

Constraints
  The following constraints apply
  • The directive may be called multiple times.
  • The directive may be called from any task.
  • The directive may obtain the object class of the timer.
  • Where the object class is not specified, the object class is the RTEMS Workspace.

Traced ICD Specification(s)
spec:rtems-timer-if-create

Traced SRS Specification(s)
spec:rtems-timer-req-create

Definition at line 212 of file timercreate.c.

```

Traced ICD Specification(s)

spec:rtems-timer-if-create

Traced SRS Specification(s)

spec:rtems-timer-req-create



Conclusion



Automated document generation
reduces costs of qualification

Automated document generation
permits frequent updates of
QDPs

Manual writing
(parts) of documents
is still required

QDPs must still be
reviewed and delta qualified



Thank You!

Email: frank.kuehndel@embedded-brains.de

Phone: +49 89 18 94 741 - 23