

# Requirements management for mission preparation at the German Space Operations Center (GSOC)

Frank Wallrapp<sup>1</sup> and Andreas Lex.<sup>2</sup>

German Space Operations Center, DLR Oberpfaffenhofen, 82230 Wessling, Germany

Verification of a contract consisting of a lot of free text and no precise requirements is a complex task. This paper describes the approach taken at GSOC during the preparation of a geostationary communication satellite mission to make this task transparent and efficient for both the customer and the operations provider. For the management of contractual requirements the commercial software DOORS together with a few specific customizations was used. All requirements and activities related to their fulfillment were stored in a central database and linked with each other. For that reason complete traceability from contract to subsystem level tests, operational procedures, inspections and design documents could be ensured continuously. Manual generation of verification matrices was rendered unnecessary. An extraction tool made it possible to provide relevant information whenever requested to the customer who was not using DOORS. This approach also enabled verification of internal requirements and tracking of the execution status of the planned tests without additional effort.

## Nomenclature

<i>CDR</i>	=	Critical Design Review
<i>DR</i>	=	Detailed Requirement(s)
<i>ECSS</i>	=	European Cooperation for Space Standardization
<i>PDR</i>	=	Preliminary Design Review
<i>ORR</i>	=	Operational Readiness Review
<i>SR</i>	=	System Requirement(s)
<i>TAR</i>	=	Technical Acceptance Review
<i>TR</i>	=	Test Report
<i>TS</i>	=	Test Specification
<i>TTP</i>	=	Test and Training Plan

## I. Introduction

**D**LR's German Space Operations Center (GSOC) is the central national institution for spaceflight operations. This responsibility includes earth observation, communication and navigation satellite missions as well as human spaceflight missions. Furthermore, contributions to astronauts training are provided, space experiments are operated and sounding rockets are developed and launched from sites around the world. Eventually, DLR works on new technologies, such as on-orbit-servicing in order to master even the most difficult space projects of the future.

The Mission Operations division fulfills the project related operational tasks associated with four main types of space missions:

- Earth Observation Satellite Missions
- Communication Satellite Missions
- Navigation Satellite Missions
- Human Spaceflight Missions .

The main tasks include the planning and preparation of missions, the definition of ground system requirements,

---

<sup>1</sup> Project Manager, German Space Operations Center, [frank.wallrapp@dlr.de](mailto:frank.wallrapp@dlr.de).

<sup>2</sup> Quality Manager, German Space Operations Center, [andreas.lex@dlr.de](mailto:andreas.lex@dlr.de).

mission planning, the generation of operation procedures as well as tests, trainings and simulations. The actual mission comprises Launch and Early Orbit Phase (LEOP) of 2-4 weeks duration and a subsequent Commissioning & Routine Phase. Professional preparation is required for a safe LEOP and the transition into the Routine Phase. The Mission Operations division also represents DLR and GSOC externally, dealing directly with the customers while supported by other divisions.

## II. Boundary Conditions

Mission operations at the GSOC are prepared on the basis of the ECSS-standards. Therefore the relevant standards have to be tailored to the project boundary conditions. In the following chapters we are focussing on the preparation of a typical communications-satellite-mission. Therefore we describe the main phases of this project-life-cycle based on the standard "ECSS-E-ST-70 C Ground systems and operations – Principles and requirements". According to ECSS-E-ST-70 C the Phases B to E are representing the time between the project-kick-off and the disposal of the satellite. The focus on this description lies on the one hand at the preparation and the management of requirements and on the other hand at the testing and the verification of the system (Phase B to D).

### A. Phases of the Project-Life-Cycle:

The project-life-cycle starts with phase A. The intention of phase A is to get a good understanding and a concept of the upcoming mission. The outcome of this phase is the basis for the contractual documents. After signing the contract Phase B starts with the kick-off-meeting. The inputs for this process are contractual documents containing technical and financial details as well as the project schedule. Especially the technical details are very important for the ground system design. phase B ends with the Preliminary Design Review (PDR) roughly 2 years before launch. The aim of this review is to present the customer the basic design of the ground system and to get a green light to proceed with the detailed design phase. This phase C contains on the one hand the generation of detailed requirements for software, facilities and external ground stations and on the other hand there is a focus on the planning of tests, trainings and simulations. Phase C ends with the Critical Design Review (CDR) typically 1.5 years before launch.

System implementation starts in phase D1 with test testing activities in phase D2. The main tasks with focus on the requirement management are the test specification, test execution and test documentation. In the Technical Acceptance Review (TAR), roughly 6 months before launch, the technical system shall be accepted and all technical requirements shall be fulfilled,

The next phase D3 before launch is focusing on operations and simulations using the completely integrated and tested ground system. Therefore in the Operational Readiness Review (ORR) all technical and operational requirements shall be fulfilled. If this is done successfully, the ground system is ready for the launch roughly six weeks later.

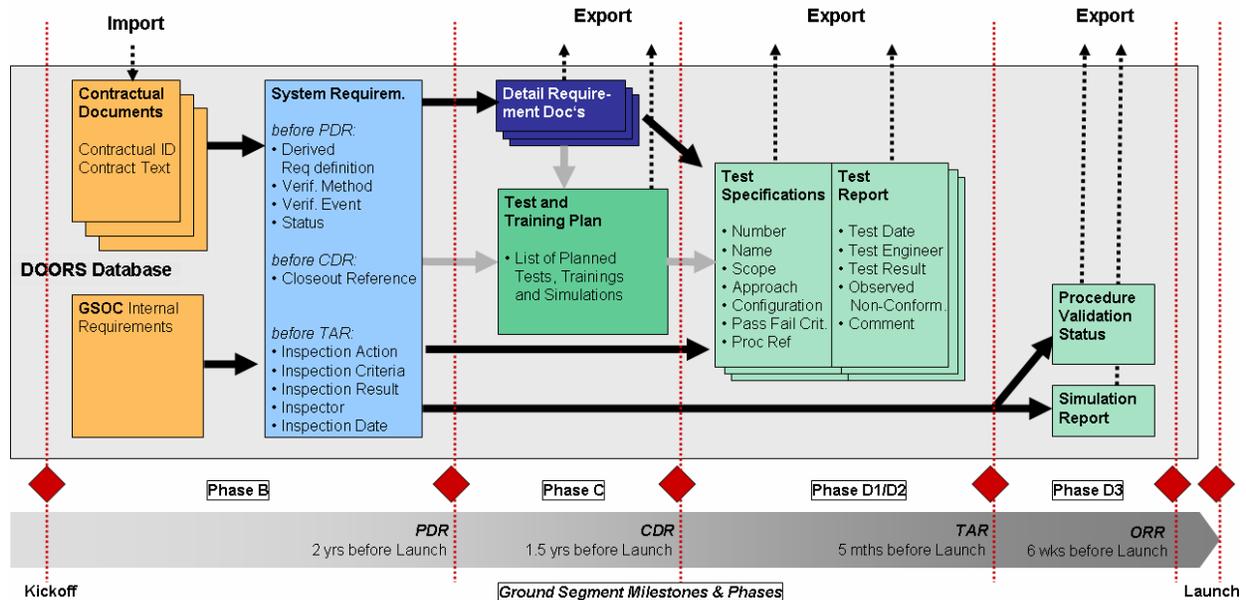
This paper covers activities performed for the preparation of a geostationary telecommunication satellite mission from Kickoff until ORR.

### B. Complex project constraints:

Complex project constraints necessitated the development of new ways to handle the contractual documents. Only a few documents contained identifiable requirements. The majority of the fifty documents was based on free text and only implicit requirements. Additionally the content of the documents was partly overlapping and partly contradictory. We therefore decided to derive internal *System Requirements* (SR) and work with these requirements in the course of the project instead of working with the contract itself. The objective was to establish a complete traceability from the requirements in the contract to the verification of system-design by using the requirements management software DOORS. This traceability should be available for our internal work and for the customer. Since the customer was not using DOORS some emphasis was also laid on import/export of Office format files.

## III. Requirements refining and tracking within the DOORS database

This chapter describes the evolution of the DOORS database in the course of mission preparation. This process will be described using Fig.1. It shows on the lower part the project phases and milestones. The upper part shows the content of the DOORS database at each of these project phases. For each milestone the items left from the marker were already contained in the database. The filled rectangles show DOORS formal modules (=data tables), the thick arrows between DOORS modules are symbolizing the DOORS links between objects (=lines in the data tables) of the modules. Thin dotted arrows show import or exports from or to the DOORS database. In this chapter we explain the activities in the different project phases by going through Fig.1 from left to right.



**Figure 1. Content of the DOORS database at the different project phases.**

*Upper Part: Content of the DOORS database. Lower Part: Project phases.*

*For each milestone the items left from the marker were already contained in the database. The filled rectangles show DOORS formal modules (=data tables) and the text inside the rectangles shows the attributes (=columns) present in the modules. The thick arrows between DOORS modules are symbolizing the DOORS links between objects (=lines) of the modules. Thin dotted arrows show import or exports from or to the DOORS database.*

### A. From Kickoff to PDR (Phase B):

#### 1. Import of Contractual Documents

The process started by importing all contractual documents into DOORS (upper orange rectangles in Figure 1). A simple approach was taken by using the standard Word to DOORS exporter and creating one DOORS module per contractual document without special preparations of the documents in advance.

#### 2. Derivation of System Requirements

Afterwards each module containing a contractual document was manually scanned through and meaningful requirements derived. These *System Requirements* (SR) were collected in one module (light blue box in Fig. 1). To allow for traceability, the SR were connected to the respective contract paragraph using the DOORS links functionality (thick arrow between Contractual Documents and System Requirements in Fig. 1). After all contractual documents were processed a review process ensured that the SR were meaningful, relevant, unique, and ordered reasonably. Since not all requirements for our work were included in the contract we added also internal requirements (lower orange rectangle in Fig. 1) to the SR list. These were treated identical to those derived from the contract. The SR module was extended and updated during the project as described in the following sections.

#### 3. Classification of "System Requirements"

At this stage of the process the following attributes were created in the SR module:

- Status
- Verification Event
- Verification Method

All SR were classified by setting these attributes. *Status* was either *Rejected* for SR we could not comply with or *Open* for SR we complied with and intended to verify. The attribute *Verification Event* determined at which milestone the requirement shall be verified and *Verification Method* how the requirement shall be verified. Possible methods were *Review of Design (RoD)*, *Inspection* and *Test*. The method *Test* was chosen not only for requirements to be verified by a classical "technical" test, but also for requirements that shall be verified by a training/simulation or by successful validation of an operational procedure\*.

\* "validation of an operational procedure" means execution of a flight or ground procedure under realistic

## **B. From PDR to CDR (Phase C)**

### *1. Derivation of detailed requirements*

In phase C Detailed Requirements (DR) with respect to software, facilities and external ground stations were derived from the SR (dark blue rectangles in Fig. 1). For each class a separate module was created and the derived DR linked to the parent SR. All documents were exported to standard office formats, because some recipients were not part of the core project team and had therefore no access to the DOORS database. Also these requirements were classified by their verification method.

### *2. Test and Training Planning*

Another important task in this phase was the rough planning of the testing and training activities. For that reason a test and training plan (TTP) was written on the basis of the SR and DR classified as to be verified by *Test* (dark green rectangle in Fig. 1). The main content of the TTP was a coarse list of planned test cases, training/simulation sessions and operational procedures along with a rough description of each event. The TTP was made available in Office Format for the customer. The SR and DR classified as being verified by *Test* were linked to the appropriate item in the TTP (Thick grey arrows ending at the TTP in Fig. 1). By doing so it was ensured that for each SR and DR that was foreseen to be verified by *Test* a proper test, training or operational procedure is identified at the end of the design phase.

### *3. System Requirements to be verified by “Review of Design”*

At CDR a first set of SR to be classified as *RoD* were taken into account in design documents. In order to allow for tracking this fact, the following attribute was created in the SR module:

- Closeout Reference

For the SR taken into account in design documents the relevant Doc No and chapter were filled into this attribute and the attribute *Status* was set to *Closed*. If a SR was verified by a design document delivered at a later review the described activity was performed at the corresponding review.

## **C. From CDR to TAR (Phase D1/D2):**

With the end of the design phase the technical implementation of the ground system started. The accompanying inspections and technical tests were also tracked by DOORS.

### *1. Inspections*

To allow tracking of inspection activities the following attributes were created in the SR module:

- Inspection Action
- Inspection Criteria
- Inspection Result
- Inspector
- Inspection Date

For each SR classified as *Inspection* these attributes were filled after the corresponding inspection was performed successfully and the attribute *Status* set to *Closed*.

### *1. Test Preparation*

The activities in conjunction with technical tests (software, hardware, interfaces) were supported and tracked by DOORS. This process can be seen roughly in Fig. 1 and in more detail in Fig. 2.

In a first step the tests coarsely defined in the TTP, were detailed in test specifications (TS) within DOORS. Various modules were created each containing one group of tests, like software tests or ground station interface tests (light green rectangles in phase D1/D2 in Fig. 1). Generally this was done by splitting each test case on TTP level in multiple test cases on TS level and linking both (Thick grey arrow starting at the TTP in Fig. 1). The created test cases were also linked to SR and DR being verified by the respective test cases (Thick black arrows ending at the TS modules in Fig. 1). This established the traceability between the requirements and the tests covering the requirements. The following attributes were created and filled with information for each test case in all test specification modules (left part of upper green rectangle in Fig. 2).

---

conditions e.g. using the satellite simulator and the integrated ground system.

- Test Name
- Test Number
- Test Scope
- Test Approach
- Test Configuration
- Pass Fail Criteria
- Test Procedure Reference

The test specification gave a rough description of the test but could not be used as a step-by-step test instruction. For that reason detailed *test procedures* were developed containing a step-by-step test instruction with expected values/behavior for each step. These procedures could cover one or more test cases. They were developed NOT in DOORS but in a spreadsheet format (lower green rectangle in Fig. 2). The link to the test case was established by putting the name of the test procedure in the attribute *Test Procedure Reference* of the relevant test cases. The DOORS module was exported and together with the referenced test procedures was delivered as “Test Specification” in office format for the customer before each test.

### 2. Test Execution

Directly before each test run a Test Review Board (TRB) was held and some general test run information was put down in a TRB protocol. The test itself was performed by going step by step through the test procedures and noting the actual value/behavior directly in the procedure spreadsheet file (see Fig. 2). By this approach the test procedure constituted also the test protocol. It has to be pointed out that all activities related to the test run itself were done manually without DOORS support. This included filling out the test procedure, filing of the procedures on the server before and after the test run as well as filing accompanying protocols, minutes of meeting and test data.

### 3. Test Reporting

After each test the information from the run was transferred into the TS DOORS module. For that purpose attributes related to the test run were added to the TS module and filled with data (right part of upper green rectangle in Fig. 2):

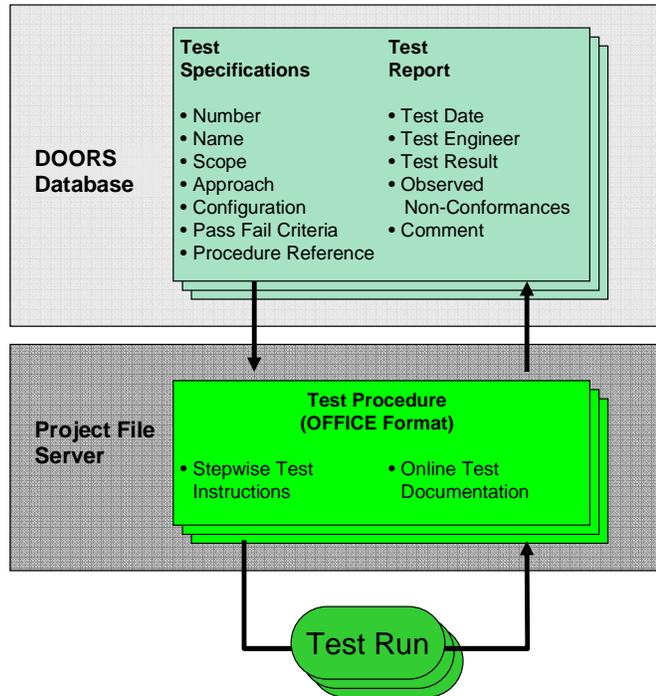
- Test Date
- Test Engineer
- Test Result
- Observed Non-Conformances
- Comment

A set of these attributes was created for each test run. By doing so, the information now contained in the DOORS module is also covering information usually contained in a test report. Consequently, an export of that information in Office format together with the filled-out as-run procedures was delivered as “Test Report” for the customer after each test run.

Although some steps in this approach were done manually outside DOORS, the traceability from the requirements to the test results could be established.

### D. From TAR to ORR (Phase D3)

After the completion of the technical tests of the ground system for the TAR, the operational activities came into focus. Also these activities were tracked inside DOORS (see Fig. 1). For that purpose two modules were created, one containing all trainings/simulations and one containing all operational procedures and their validation details



**Figure 2. Testing process.** DOORS modules were created that hold detailed information of tests (upper rectangles). One module contained one group of tests. Before the test run, only information required for a test specification was contained. After the test run also information required for a test report were filled in. Stepwise test instructions were created in office format and also served as online test documentation. They were referenced in the DOORS module.

(light green rectangles in phase D3 in Fig. 1). The list of simulations/trainings as well as the list of operational procedures was exported for the customer before ORR.

#### 1. *Tracking of Simulations and Trainings*

Simulations and training were treated as follows: After each simulation/training, one object (=line) was added to the respective DOORS module containing the following information: Date, Goal, Content, Participants, and Outcome. For some special events, like the large rehearsal with the spacecraft manufacturer a plan was written before the event, which was also referenced in the DOORS module. SR that were verified with one of these events were linked to the events (thick black arrow between System Requirements and simulation report in Fig. 1) to enable traceability.

#### 2. *Tracking of Operational Procedure Validation*

The approach for operational procedures was slightly different. Here, we prepared the list of ground and flight operations procedures before the validation activities started and monitored the details of all validation runs. SR that were verified with one of these procedures were linked to them (Thick black arrow between System Requirements and procedure validation status in Fig. 1) to enable traceability.

### **IV. The Benefit: Generation of verification matrices**

In the previous chapter we described the approach we took to build up a complex DOORS database filled with modules containing the contract, derived requirements, and test and training documents. We also described the establishment of DOORS links between objects of these modules. In this chapter we report the main benefit of this approach: The ability to run analysis's generating matrices giving the actual verification status using the structure in the database. First we give some general information on this topic then we discuss in more detail the contractual verification matrix we generated for the customer at every review.

#### **A. General**

The big advantage of using a requirements management tool like DOORS is the ability to collect information from various modules in one well-arranged traceability matrix. This "analysis" functionality is established by the links between objects contained in different modules. Out-of-the-box DOORS contains some possibilities to define such analysis's, which can be customized using the DOORS macro language DXL. In our project we created the following traceability matrices:

1. Detailed Requirements Verification
2. Test Tracking
3. Contractual Verification

The first one consisted of the list of detailed requirements from the respective modules in one column and the status (Results & Date) from all tests linked to the corresponding requirement in the other column.

The second one consisted of the list of planned tests from the TTP in one column and the status (Result & Date) from all linked detailed test cases in the other column.

Both of these matrices were generated for internal tracking purposes. While these two matrices could be generated with the standard DOORS analysis feature which simply adds information from all linked objects to the matrix, the third one was rather complex and was generated using a customized DXL script including processing of the linked data before filling the matrix. This script enabled:

- Removal of duplicate entries in linked information
- Generation of summary values from linked information
- Numbered output of linked information
- Collection of data following two or more subsequent links.

The content of the third matrix shall be described now.

#### **B. Contractual Verification Matrix**

##### 1. *General*

One main purpose of our approach was to be able to present at each major project milestone the status of the contractual requirements verification to the customer. Four verification matrices were generated based on the four contractual documents for which the customer requested a detailed verification. They were generated at CDR, TAR and ORR each time containing more information. The content of the matrices at each review is an excerpt of the DOORS database content available at that review and can be seen schematically in Fig. 3a. The DOORS matrices were exported in Spreadsheet format for the customer. Fig. 3b and c show exemplary screenshots from one matrix delivered at ORR.

In general, one line of one matrix consisted of one paragraph of the contractual document in the first column with information from linked objects of other modules in the other columns. Please note that the colors on top of the columns in Fig.3 are indicating from which DOORS module the information is coming from. The colors are identical to the corresponding modules from Fig. 1.

## 2. Matrix at CDR

At CDR, the matrices contained in the first column the contractual text (orange in Fig. 3a,b) and in the following columns attributes of the linked SR (blue in Fig. 3a,b). Some attribute values from the linked SR were summarized in the matrices using the customized DXL script collecting the information. For example, if several of the linked SR have the same attribute value, it is only given once in the matrices. For some columns, the values of the linked SR were evaluated to give only a single entry per contract paragraph. As an example, if all of the linked SR have the attribute *Status* set to *Closed* then the column *Verification Status* in the matrices was filled with “OK”. If one of the linked SR does not have a *Closed* status, then the entry was “P”. If none of the linked SR is *Closed* then the entry was “—”.

The last column at this project phase contained the references to all the planned tests from the TTP (dark green in Fig. 3a,b) that were linked to those SR linked to the contract paragraph. Please note that in this case information which is “two links away” from the contract has been collected, as can be seen in Fig. 1. The matrix at this stage contained the columns A to G from Fig. 3b.

With these matrices the customer was able to check our approach of verifying contractual requirements at the end of the design phase. For some requirements that were already verified by design documents at CDR, the corresponding reference was filled in the attribute *Closeout Reference* at SR level and appeared in the corresponding matrices column, as can be seen in the upper line of Fig. 3b. At this stage the entry *Verif. Status* was different to the ORR status shown in Fig. 3b. The value was “P” indicating a partly verified contractual text for the upper line in Fig.3b and “—” for the lower indicating a not verified contractual text, since the verifications were due for TAR and have not been performed yet at CDR.

## 3. Matrix at TAR

At TAR, the technical tests were completed and the details of the tests were documented in the test specification/report modules. By using the DOORS link structure the details of the tests (Date, Result, Test Report-No, Test No) from the test specification/report modules were put as an additional column into the matrices. This is schematically shown by the upper part of the light green column in Fig. 3a and column H in Fig. 3b. With these entries, the customer was able to check the status of the tests associated with a paragraph from the contract at a glance. Details of the tests could be traced by reading the referenced Test Report and Test Number.

As written in section III-C-1, details of inspections were documented in the SR module. All those attributes associated to inspections were put in separate columns into the matrices for the TAR delivery. This is schematically shown in the lower part of the blue column in Fig. 3a and exemplary in Fig. 3c which shows the columns I to M of the matrices containing information about inspections. As at CDR, for requirements verified by design documents the corresponding reference was filled in the attribute *Closeout Reference* at SR level which was also part of the matrices.

## 4. Matrix at ORR

At ORR all entries corresponding to design documents, inspections and technical tests were completed. Additionally, for those SR that were verified with a training, simulation or procedure validation, the corresponding details of the event were added to the test details column. This was accomplished by following the links from the contract to the entries in SR module and further to the procedure validation and training simulation module. This is schematically shown in the lower part of the light green column of Fig. 3a. With the ORR delivery of the matrices the customer was able to finally check how the requirements to be met before launch were verified. These matrices built the basis for the customer to give the go for Launch.

Contract Document	System Requirements (in separate columns)	Test / Training Plan References	Test and Training Details (in one column)
<b>Since CDR</b> - Contractual ID - Contract Text	<b>Since CDR:</b> - Compliance Statement - Verif. Event - Verif. Method - Closeout Reference - Verif. Status  <b>Since TAR:</b> - Inspection Results - Inspection Criteria - Inspection Action - Inspector - Inspection Date	<b>Since CDR:</b> - Planned Test Case from TTP	<b>Since TAR</b> - Test Spec / Report No. - Test Number - Result (per run) - Date (per run)  <b>Since ORR</b> - Procedure Validation Details - Simulation References
•...	•...	•...	•...

**Figure 3a. Schematic layout of the contractual verifications matrix.** One line in the matrix consists of one paragraph from the contract in the first column and information from various linked modules in the other columns. The matrix was extended to contain more columns from review to review.

	A	B	C	D	E	F	G	H
1	Contract Text	Comp.	Verif. Event	Verif. Method	Closeout Reference	Verif. Status	TTP References	Test Details
2	SPS 321 A Satellite Simulator must be available in all Mission Phases	C	1: CDR 2: TAR	1: RoD 2: Test	Sim Integration described in BS-0062(1.4), § 5.2.4	OK	- PL-0061: TC 9.1-140: Sim tests	- PL-0502/PN-0012: I-SCC-RFB-1.4.1: Passed (23 October 2008) - PL-0502/PN-0012: I-SCC-RFB-1.4.2: Passed (23 October 2008) - PL-0502/PN-0012: I-SCC-RFB-1.4.3: Passed (23 October 2008)
3	SPS 322 The control room must be equipped with 12 consoles	C	TAR	Inspection		OK		

**Figure 3b. Exemplary screenshot of the contractual verification matrix as exported for the customer at ORR, columns A to H.**

	A	I	J	K	L	M
1	Contract Text	Inspektion Results	Inspektion Criteria	Inspektion Action	Inspector	Inspektion Date
2	SPS 321 A Satellite Simulator must be available in all Mission Phases					
3	SPS 322 The control room must be equipped with 12 consoles	The control room contains 12 consoles	The number of consoles	Count number of consoles	John Doe	09.09.2009

**Figure 3c. Exemplary screenshot of the contractual verification matrix as exported for the customer at ORR, columns A and I to M.**

## V. Summary / Conclusion

The process described in this paper was the first effort made at GSOC to use the requirements management tool DOORS in a satellite operations project. After a successful execution of LEOP and a few months of routine operations, we came to the conclusion that the introduction of such a tool was of great benefit for the mission preparation. It simplified tracking of the requirements internally and the reporting externally. In a next step it is planned to integrate the activities in connection with the test execution into DOORS to eliminate the manual activities connected to testing as described in chapter III-C-2. However even at the current stage, the process described in this paper is the basis for other satellite missions currently at preparation at GSOC that also are using DOORS.