DPE

„Process documentation“

For

KCC

Vienna, 07/29/2008

ComSolution GmbH

Print Date: 22. August 2008 11:52:00

Document-Id: 2008-07-30.jol.processdocumentation.v04.doc

Author: Johannes Lorenz

Preface

This document represents the central output of phase II. It is a compendium of all process-designs and related definitions which have been developed by the KCC-project “DPE”. It contains the reviewed and accepted designs and is intended as the basic input for the pilot projects to be carried out in phase III.

This document has been compiled on the basis of data and information which initially has been collected in the process-designer of process4.biz.

The document is structured according to the process-map (1.1) and shows an analog sub-structure for any of the relevant primary processes. This sub-structure contains as well a graphical representation of the respective primary process and its secondary process as a textual description.

In the last chapter of this document (7) one can find a dictionary containing definitions and explanations of the relevant terms and expressions.

Contents

1 Overview

This chapter of the document at hand describes the process-design at the level of primary processes. Using a process-map (see 1.1) the big picture of the new design is given. Any of the relevant primary processes is covered in a separate chapter of this document. The respective secondary processes are as well described there using detailed RACI-charts.

1.1 Section of Process-map

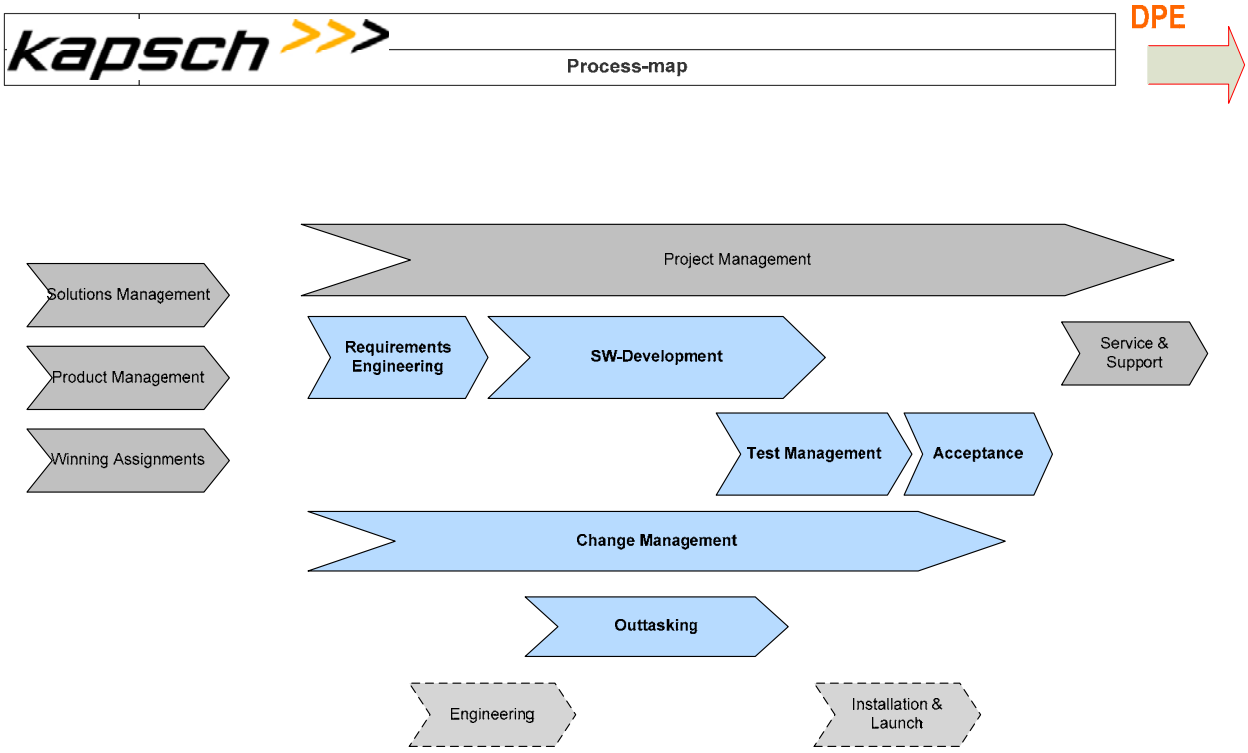


Exhibit 1: Section of the KCC process-map

The section of the KCC process-map which has been relevant for the process redesign is displayed in Exhibit 1. Those processes which have been object of the redesign itself are marked blue and those which served as constraints are grey. (Two items of the last group are displayed with dashed lines indicating the outdated definition of these processes.)

The process-map delivers an overview of how the individual processes are interrelated. Starting on the left side, one can find the three processes – Solutions Management, Product Management and Assignment winning – which are initiating projects and therewith software-development. The process project management is exhibited on the top of the map, as it is seen as the one controlling or triggering all the others. The three other processes – Service & Support, Installation & Launch and Engineering – are basically receivers of inputs (from the blue ones).

1.2 Scope

Scope of the project DPE has been to rework the following processes:

Requirements Engineering

Software-development (including Acceptance)

Test-Management

Change-Management

Outtasking

All these are described within this document using a separate section (2 through 4) for any of them. This includes the definition of the individual inputs, outputs, activities, roles and responsibilities.

The definition of metrics is not part of this document. These are defined in the document “DPE_Metrics for KCC”. As well the designs of the various templates, checklists and guidelines are not provided here but in specific separate documents.

2 Requirements-Engineering (RE)

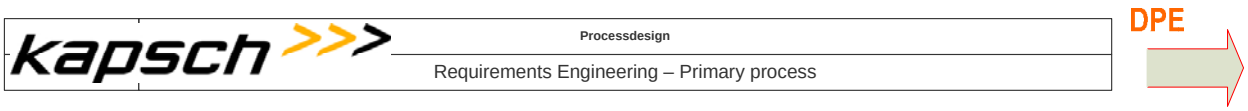
Requirements Engineering is the process which organizes the activities from an initial trigger (e.g. a request for proposal) to a consistent and complete set of technical requirements – the TRS. Main goal of the process is to manage all requirements in a way which ensures traceability from the basic documents to the deliverables (URS, TRS and design draft) of the process.

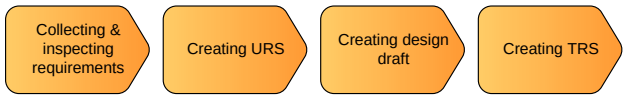
The process is strongly related to the processes of Winning Assignments, Product Management and Project Management.

The execution of the process is supported by a tool with the working title “RE-DB” (i.e. requirements engineering database).

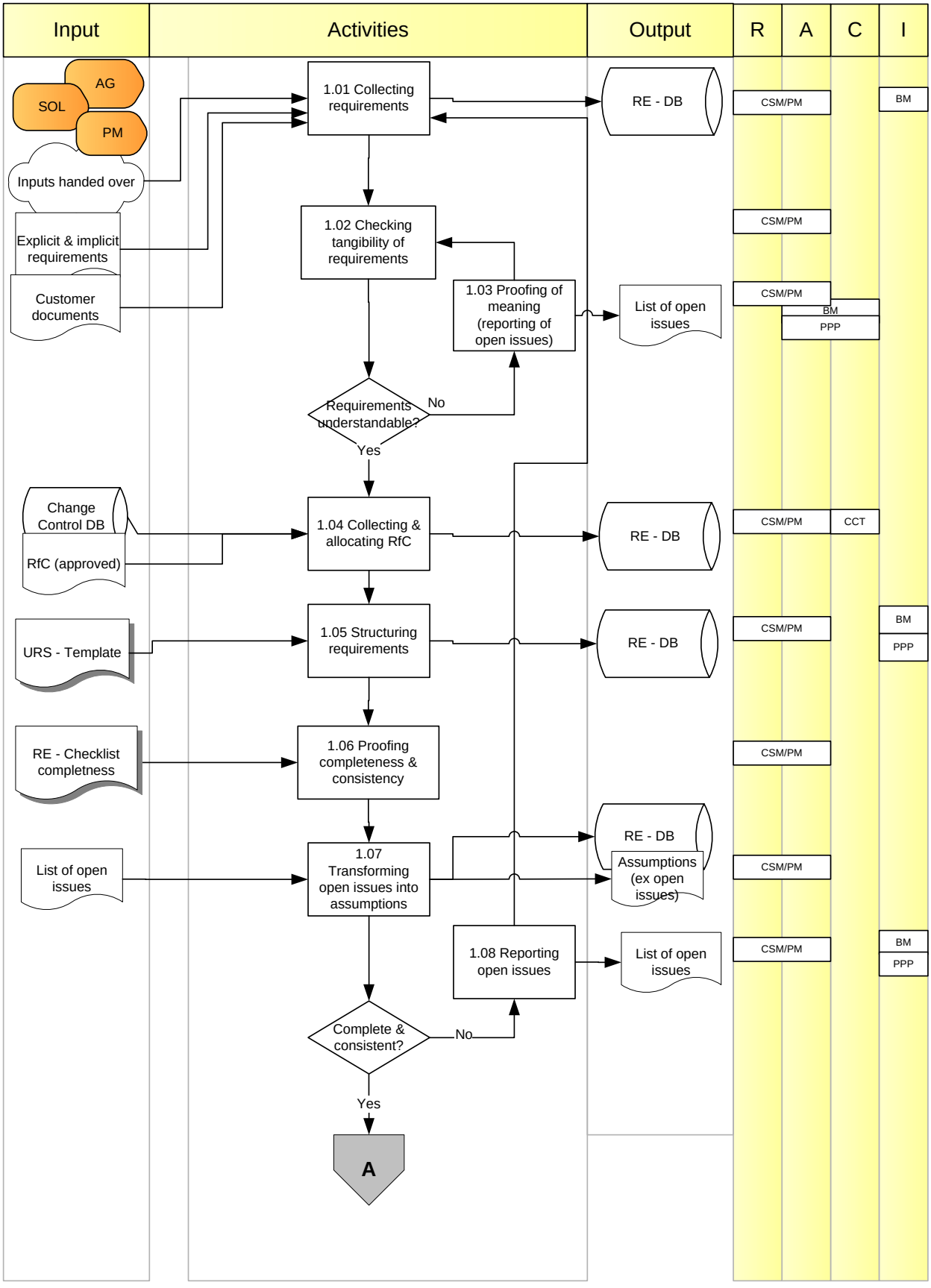
2.1 Primary process

On this level Requirements Engineering is decomposed into four parts (secondary processes): Collecting & inspecting requirements, Creating URS, Creating design draft and Creating TRS.





2.2 RE – Collecting & inspecting requirements



2.2.1 Core activities

2.2.1.1 Activity 1.01 Collecting requirements

This activity covers the systematic entering of requirements of all kinds into the RE-DB. Sources could be customer documents, workshops, market researches and minutes of customer contacts. Besides any source of implicit requirements could be relevant.

Requirements are entered according to a specific template, which defines the set of attributes connected to the entity “requirement”.

2.2.1.2 Activity 1.02 Checking tangibility of requirements

As it is highly important that the once entered requirements are clearly defined, the goal of this activity is to ensure if the individual requirements are understandable apart from any other information or data.

2.2.1.3 Activity 1.03 Proofing of meaning (Reporting open issues

In case the decision “requirements understandable” delivered a negative response it is the responsibility of the CSM/PM to clarify the open issues or report them in case clarification is not possible directly.

2.2.1.4 Activity 1.04 Collecting & allocating RfC

This activity serves as an interface to the process of Change Management. In case relevant RfC are stored in the Change Control DB, it is the task of this activity to integrate these RfC into the actual set of requirements.

2.2.1.5 Activity 1.05 Structuring requirements

The task of this activity is to create a well arranged structure within the actual set of requirements. A hierarchical structure has to be created, which conforms to the URS-Template and already prepares a later structure used in the design draft.

2.2.1.6 Activity 1.06 Proofing completeness & consistency

The task of this activity is to check whether the actual set of requirements is sufficient for description of the solution to be created. As well a consistency check is performed. This proofs the fact that none of the requirements is contradictory to another. This task is supported by a checklist (see 2.2.2.5).

2.2.1.7 Activity 1.07 Transforming open issues into assumptions

At this activity the latest point in time is reached where the list of open issues has to be emptied. Open issues are translated into assumptions (i.e. especially marked requirements) to the extent this is possible.

2.2.1.8 Activity 1.08 Reporting open issues

If consistency or completeness are not achievable, open issues are reported and a return to the beginning of the process is performed.

2.2.2 Essential documents

2.2.2.1 Customer documents

Depending on the individual customer different qualities of initial documents are possible.

The following types of documents are possible inputs to the RE process:

Needs the target system has to fulfill

Request for proposal

Product descriptions

Contractual documents

Graphical representations on any level of abstraction

2.2.2.2 List of open issues

Structured list of issues to be clarified until a certain date or certain point of the process. Open issues are documented using the respective template.

2.2.2.3 RfC (approved

See 3.3.

2.2.2.4 URS – Template

See 2.3

2.2.2.5 RE - Checklist completeness

The checklist enables the CSM/PM to find an objective decision regarding completeness and consistency.

2.2.2.6 Assumptions (ex open issues

Assumptions are especially marked requirements documented within the RE-DB.

2.2.3 Roles

2.2.3.1 BM

Bid Manager, see Process definition “Winning Assignments”

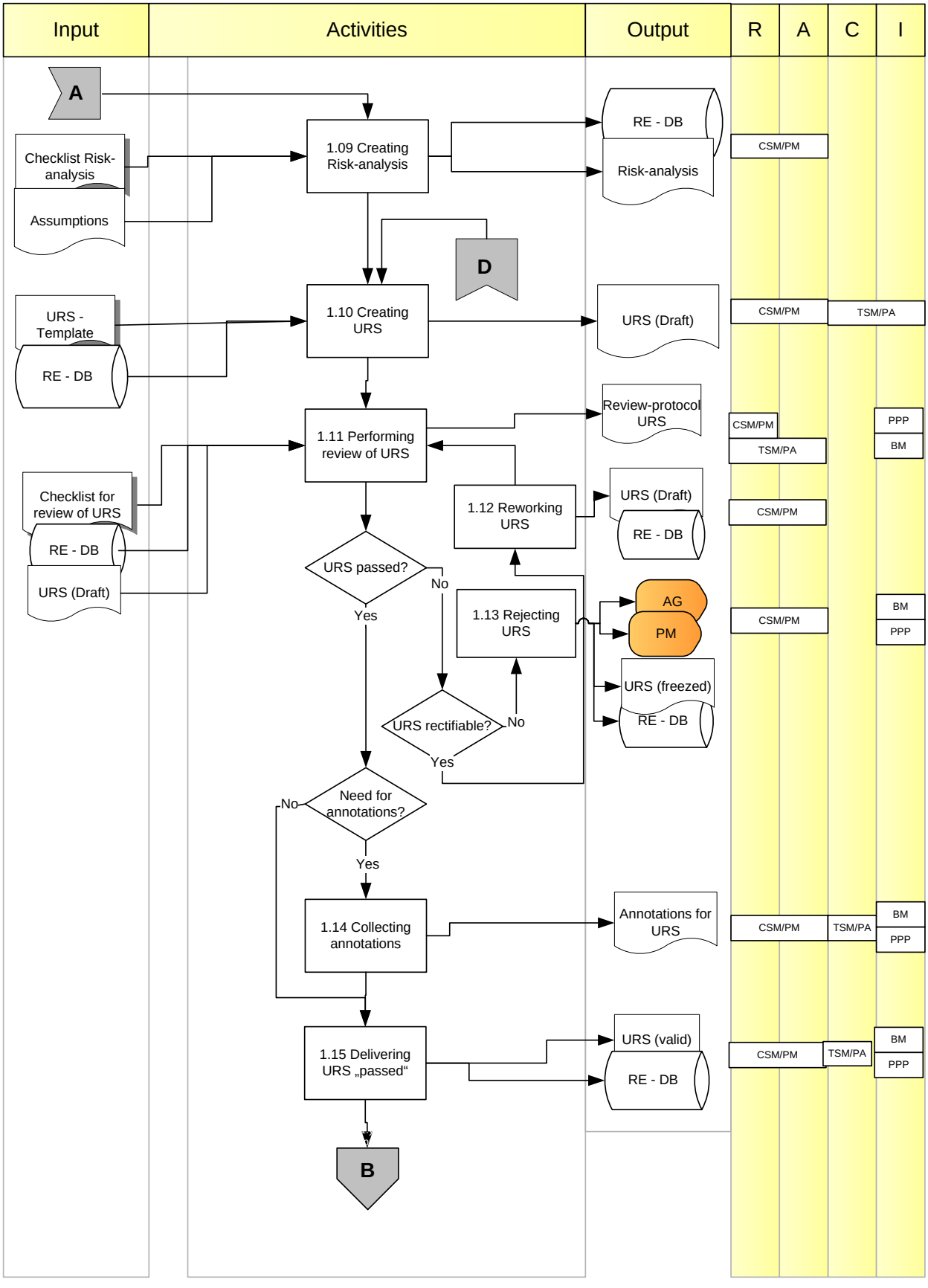
2.2.3.2 CCT

Change Control Team, see 3.

2.2.3.3 PPP

Product Policy Platform, see ??.

2.3 RE – Creating URS



2.3.1 Core activities

2.3.1.1 Activity 1.09 Creating Risk-analysis

The risk-analysis is conducted for any requirement delivering individual ratings for each of them. The risk-analysis is performed on the basis of a specific checklist (see 2.3.2.5) and stored into the RE-DB.

2.3.1.2 Activity 1.10 Creating URS

Based on the respective template the URS is generated. Technically the URS is a report from the data stored in the RE-DB. The URS is handled by the version control.

2.3.1.3 Activity 1.11 Performing review of URS

The review of the URS is performed by CSM/PM and TSM/PA of which the latter have the authority of passing or rejecting the URS. The review is documented using the respective template.

2.3.1.4 Activity 1.12 Reworking URS

This activity represents the task of correcting the URS, in case this is necessary.

2.3.1.5 Activity 1.13 Rejecting URS

If the URS is not rectifiable it is rejected by the responsible TSM/PA to the processes of AG (i.e. Winning assignments) or PM (i.e. Product management) respectively.

2.3.1.6 Activity 1.14 Collecting annotations

This activity is being carried out, if the URS has passed the review with some remarks from the side of the reviewers. “Passed with annotations” implies that the corrections have been drafted during the review and need only to be realized by the CSM/PM. In this case no further review (of this version) of the URS is necessary.

Annotations are documented by means of a simple template or directly stored into the RE-DB.

2.3.1.7 Activity 1.15 Delivering URS „passed“

This activity represents the formal closing of the creation of the URS – the valid version is handed back to the initiating process.

2.3.2 Essential documents

2.3.2.1 Annotations for URS

Work product of activity 1.14

2.3.2.2 Assumptions

Input to the risk-analysis.

2.3.2.3 URS (freeze)

In case the URS is rejected, it is stored into the RE-DB as a not changeable version.

2.3.2.4 URS (valid)

The version of the URS which has passed the review

2.3.2.5 Checklist Risk-analysis

Base document for the risk-analysis; see respective template.

2.3.2.6 RE - DB

Central tool for the requirements engineering process.

2.3.2.7 Review-protocol URS

Evidence of the review of the URS; compiled on the basis of the respective template.

2.3.2.8 Risk-analysis

Result of activity 1.09; created on the basis if a specific template.

2.3.2.9 URS - Template

Boilerplate which is used for the creation of the URS

2.3.2.10 URS (Draft

Version of the URS before it is reviewed

2.3.2.11 Checklist for review of URS

Manual for the execution of the review of the URS.

2.3.3 Processes

2.3.3.1 AG

Winning assignments, see respective definition on intranet

2.3.3.2 PM

Product management, delivering the relevant product roadmap; see respective definition on intranet

2.3.4 Roles

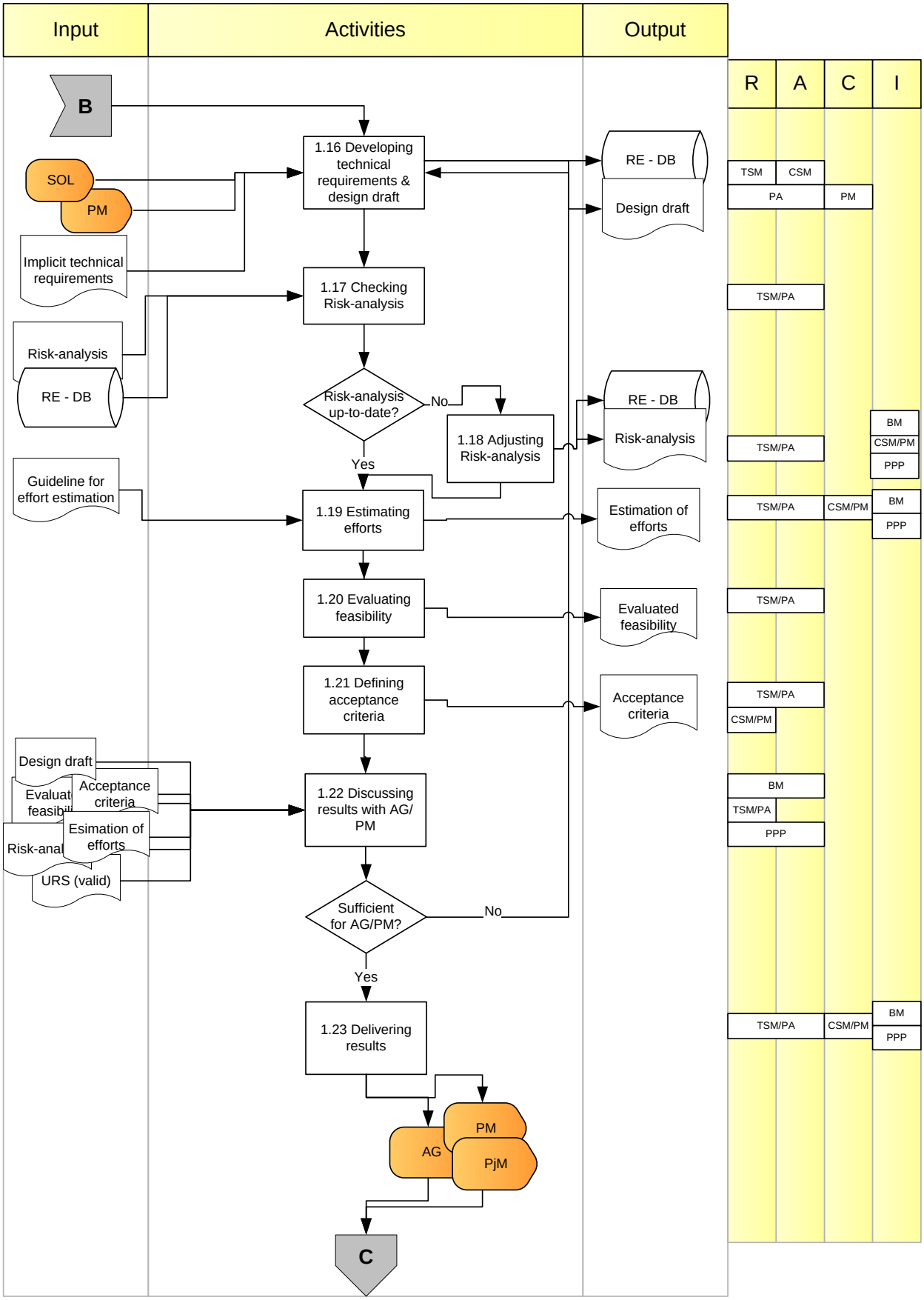
2.3.4.1 CSM/PM

Customer Solution Manager / Product Manager

2.3.4.2 TSM/PA

Technical Solution Manager / Product Architect

2.4 RE – Creating design draft



2.4.1 Core Activities

2.4.1.1 Activity 1.16 Developing technical requirements & design draft

The task of this activity is to create a preliminary high level design based on the URS.

This design draft is a graphical representation of a set of requirements with a technical focus. Whereas the URS was focused on requirements seen from a customer's perspective, this activity includes all requirements from all stakeholders (i.e. functional, non functional, implicit).

2.4.1.2 Activity 1.17 Checking Risk-analysis

As the risk-analysis may need adaptations after the design draft has been created, this activity is responsible for a proof of any impacts from this side.

2.4.1.3 Activity 1.18 Adjusting Risk-analysis

The task of this activity is to update the risk-analysis if necessary.

2.4.1.4 Activity 1.19 Estimating efforts

This activity delivers a guideline-based estimation (see 2.4.2) of the work needed for a possible realization project.

2.4.1.5 Activity 1.20 Evaluating feasibility

The task of this activity is to evaluate the technical feasibility of the possible project. As well the availability of resources and skills is checked.

2.4.1.6 Activity 1.21 Defining acceptance-criteria

According to the V-Model it is necessary to define criteria for acceptance as soon as possible. This task is performed on the basis of the respective template (see 2.4.2.4) and documented in a section of the design draft.

2.4.1.7 Activity 1.22 Discussing results with PM/AG

The task of this activity is to create a common understanding of all outputs available so far.

2.4.1.8 Activity 1.23 Delivering results

This activity represents the hand-over of responsibility to other processes.

2.4.2 Essential documents

2.4.2.1 Guideline for effort estimation

This guideline is used to carry out an estimation of efforts by means of an "experts estimate" (empirical estimation procedure) focused on the work per requirement/function in personnel days and/or monetary units

2.4.2.2 Implicit technical requirements

Sources of requirements that are not documented in the official base documents. Implicit requirements are often “no brainers” but still need to be documented!

2.4.2.3 Design draft

The design draft displays the specific set of requirements in a graphical view.

2.4.2.4 Template Acceptance criteria

This template supports the definition of acceptance criteria. These represent the highest granularity of requirements to the acceptance procedure. It includes the definition of parameter and to be measured key indicators.

2.4.3 Processes

2.4.3.1 SOL

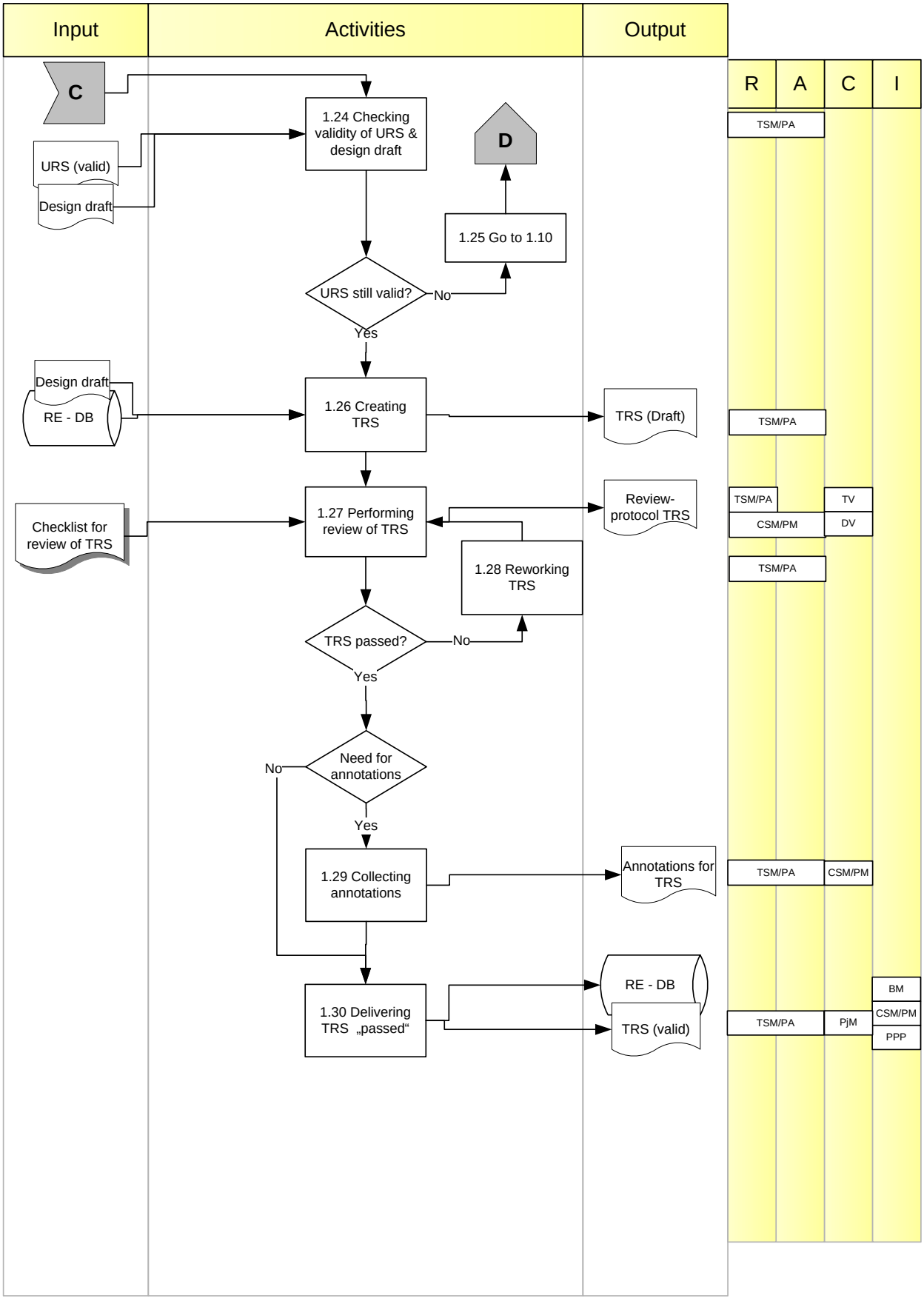
Solutions Management; see respective definition on intranet

2.4.4 Roles

2.4.4.1 PPP

Product Policy Platform; see respective definition on intranet

2.5 RE – Creating TRS



2.5.1 Core activities

2.5.1.1 Activity 1.24 Checking validity of URS & design draft

As there could be a certain time-lag between the finalization of the URS and the creation of the TRS, this activity is performed. Hence, the main task here is to verify if the URS and the design draft are still up to date.

2.5.1.2 Activity 1.26 Creating TRS

Based on the respective template the TRS is generated. Technically the TRS is a report from the data stored in the RE-DB. The TRS is handled by the version control

2.5.1.3 Activity 1.27 Performing review of TRS

The review of the TRS is performed by TSM/PA and CSM/PM of which the latter have the authority of passing or rejecting the TRS. The review is documented using the respective template.

2.5.1.4 Activity 1.28 Reworking TRS

This activity represents the task of correcting the TRS, in case this is necessary

2.5.1.5 Activity 1.29 Collecting annotations

This activity is being carried out, if the TRS has passed the review with some remarks from the side of the reviewers. “Passed with annotations” implies that the corrections have been drafted during the review and need only to be realized by the TSM/PA. In this case no further review (of this version) of the TRS is necessary.

Annotations are documented by means of a simple template or directly stored into the RE-DB

2.5.1.6 Activity 1.30 Delivering TRS „passed

This activity represents the formal closing of the creation of the TRS – the valid version is handed back to the initiating roles/processes.

2.5.2 Essential documents

2.5.2.1 Annotations for TRS

Work product of activity 1.29 (see 2.5.1.5)

2.5.2.2 TRS (Draft

Version of the TRS before it is reviewed

2.5.3 Roles

2.5.3.1 DV

Head of design (in German “Design-Verantwortlicher”)

2.5.3.2 TV

Head of test (in German “Test-Verantwortlicher”)

3 Change Management (CM)

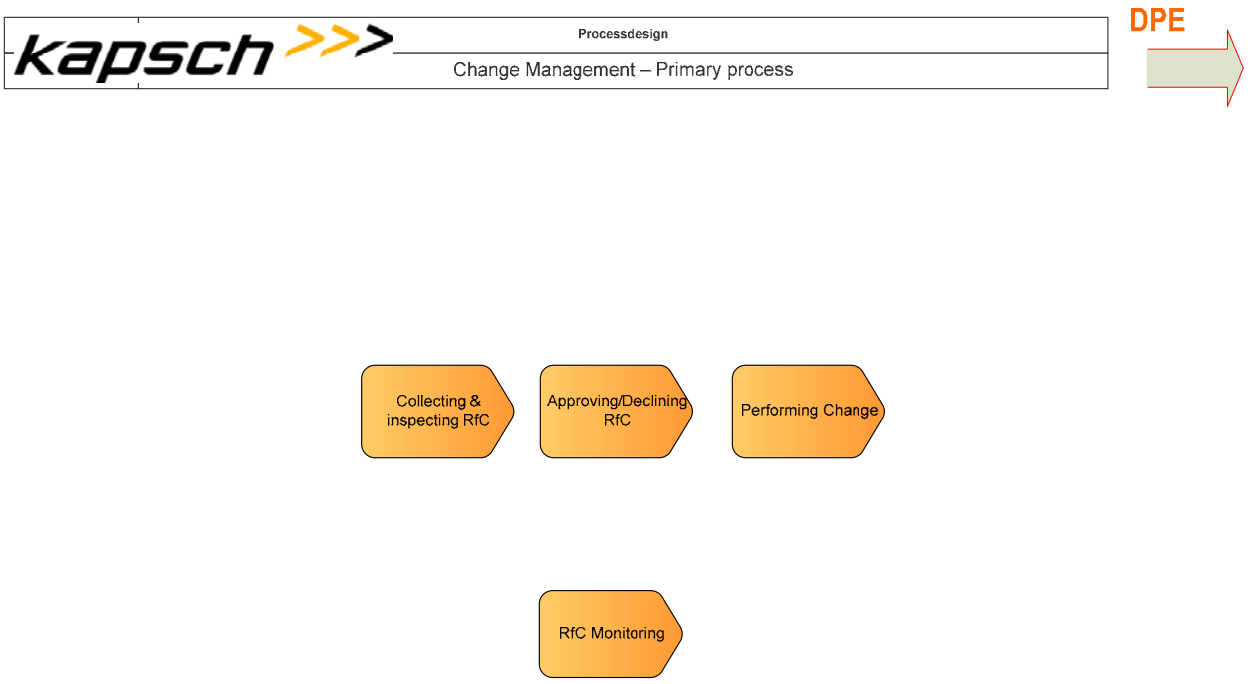
Change Management is the process which organizes the activities concerning the handling of Requests for Change (RfC). The main goal of the process is to pass through all RfC in an efficient and effective way, ensuring transparency of decisions and actions.

The process is linked to Incident and Problem Management on the one hand and Requirements Engineering and Project Management on the other.

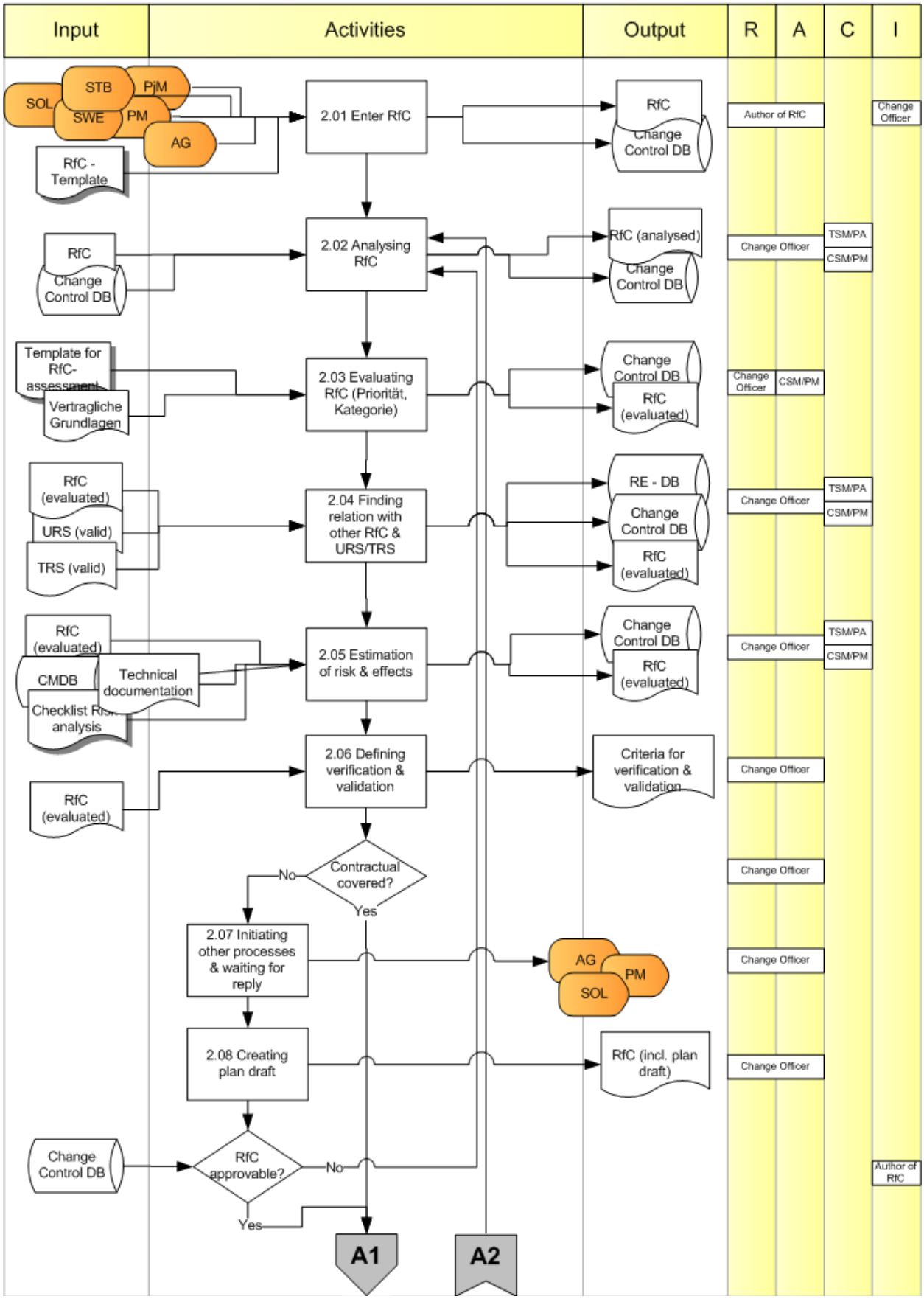
The execution of the process is supported by a tool with the working title CC-DB (i.e. change control database).

3.1 Primary process

On this level Change Management is decomposed into three linearly connected (Collecting & inspecting RfC, Approving/Declining RfC and Performing Change) and one periodically performed process called RfC Monitoring.



3.2 CM – Collecting & Inspecting RfC



3.2.1 Core activities

3.2.1.1 Activity 2.01 Entering RfC

An RfC could be entered by any person with a user in the Change Control Database. There a respective screen, which is an implementation of the RfC-Template (see 3.2.2.1) has to be used. Already there a prequalification regarding priority and category is entered.

3.2.1.2 Activity 2.02 Analyzing RfC

According to an automatic determination the RfC is assigned to a specific Change Officer (see 3.2.4). This allocation is based on the subject (i.e. the target area) and the priority of the RfC. In case, the allocation could not be performed automatically, the RfC is assigned to the “Default Change Officer” (i.e. Screener).

3.2.1.3 Activity 2.03 Evaluating RfC (Priority, Category

The task of this activity is to determine priority and category of the specific RfC. The priority (low, medium, high, urgent) is assigned according to the impact of the respective problem. This assigns the level of urgency to the RfC and results in an order of processing, if more than one RfC are open in parallel.

The category on the other hand is determined according to the nature of the proposed change. Possible categories are “Major” (large impact), “Significant” (medium impact), and “Minor” (small impact) and “Standard” (preauthorized change that has a small impact).

3.2.1.4 Activity 2.04 Finding relation with other RfC & URS/TRS

The task of this activity is to determine if other (not closed) RfC or an actual URS/TRS have any sort of interdependence with the current RfC. This evaluation results in a respective documentation in the CC-DB.

3.2.1.5 Activity 2.05 Estimating risk & effects

Based on the checklist (see 3.2.2.6) for risk-analysis this evaluation is performed.

3.2.1.6 Activity 2.06 Defining verification & validation

The task of this activity is to define methods to be performed for validation (i.e. tests) and verification (i.e. reviews). A specific template supports this activity (3.2.2.4).

3.2.1.7 Activity 2.07 Initiating other processes & waiting for reply

In case the contractual basis is missing or not clear, other processes triggered.

3.2.1.8 Activity 2.08 Creating plan draft

This activity delivers a drafted plan for implementation of the RfC. After this activity has been completed, it is checked if the RfC could be approved.

3.2.2 Essential documents

3.2.2.1 RfC

For the registration of an RfC a specific template/screen is used.

3.2.2.2 RfC Evaluation

Further templates/screens are used for evaluation, approving, closing etc. of the RfC

As well the risk-analysis is part of the RfC evaluation.

3.2.2.3 Technical documentation

All sorts of sources which are used for the risk-analysis.

3.2.2.4 Criteria for validation & verification

A template that supports the definition of the respective criteria.

3.2.2.5 Contractual basis

All sorts of documents that are basis for the relevant customer relation.

3.2.2.6 Checklist Risk-analysis

Base document for the risk-analysis; see respective template

3.2.3 Processes

3.2.3.1 STB

Process of “Störungsbehebung”, see respective definition on intranet.

3.2.3.2 SWE (=SWD)

Process of SW-Development, see chapter 4 of this document.

3.2.4 Roles

3.2.4.1 Change Officer

Responsible role for the administration of a single RfC.

3.2.4.2 Author of RfC

Any person who has the right to enter an RfC in the CC-DB.

3.2.5 Tools

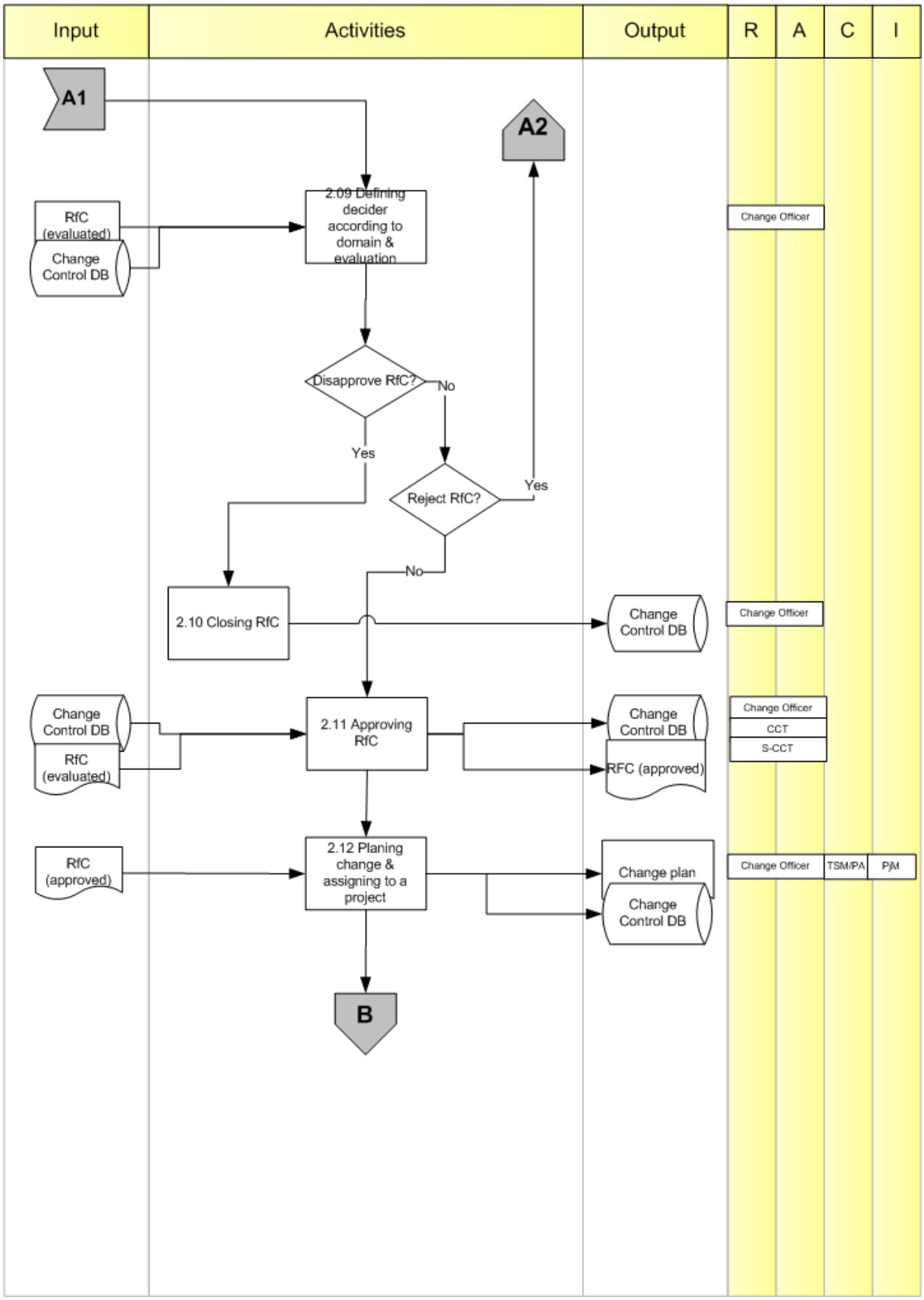
3.2.5.1 Change Control DB (CC-DB)

Tool that supports the process of change management.

3.2.5.2 CMDB

Configuration Management Database. A repository of information related to all the components of an information system. In the ITIL context, a CMDB represents the authorized configuration of the significant components of an IT environment.

3.3 CM – Approving/Declining RfC



3.3.1 Core activities

3.3.1.1 Activity 2.09 Defining decider according to domain & evaluation

Based on the actual evaluation the decider is defined. Depending on domain, category and risk-analysis the Change Officer himself or the CCT or the Senior CCT is authorized to approve the RfC.

3.3.1.2 Activity 2.10 Closing RfC

In case the RfC is disapproved it is closed formally including an explanatory statement.

3.3.1.3 Activity 2.11 Approving RfC

In case all prerequisites are satisfied, the decider approves the RfC.

3.3.1.4 Activity 2.12 Planning change & assigning to a project

After approval a detailed change plan is elaborated and the RfC is assigned to the appropriate project.

3.3.2 Essential documents

3.3.2.1 Change plan

Like the “Forward Schedule of Changes” in the ITIL context, the change plan contains details of the approved change and its proposed implementation date.

3.3.3 Roles

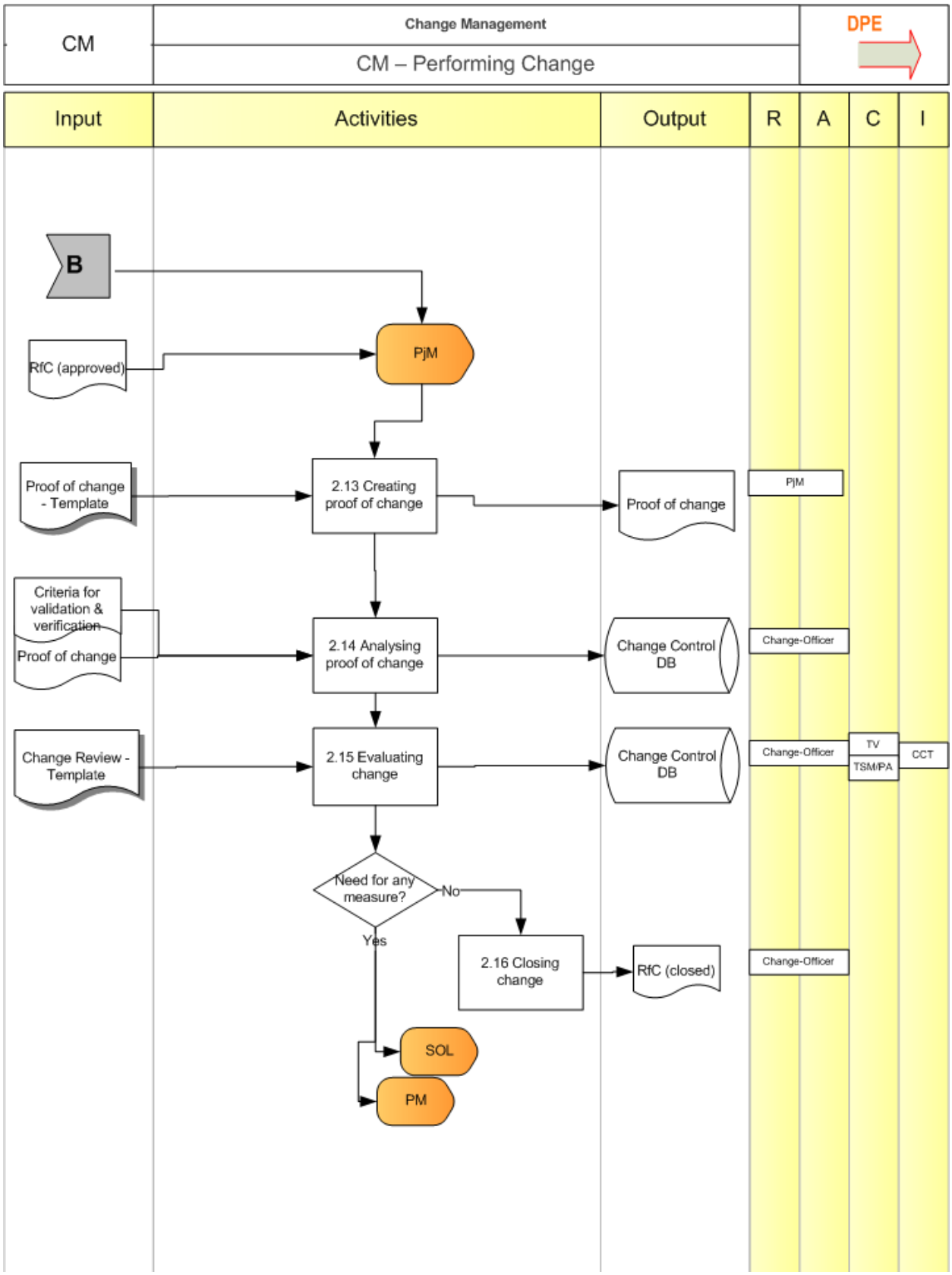
3.3.3.1 CCT

Change Control Team; dynamic group of people (depending on the change) that approves changes with medium to high rating.

3.3.3.2 S-CCT

Escalation level of the CCT; as a general rule represented by the senior management (1.ME)

3.4 CM – Performing Change



3.4.1 Core activities

3.4.1.1 Activity 2.13 Creating proof of change

The change itself is performed by the assigned project and thereafter this activity evaluates if the change was successfully or not. This proof is performed on the basis of the respective template.

3.4.1.2 Activity 2.14 Analyzing proof of change

Using the verification and validation criteria the proof of change is analyzed.

3.4.1.3 Activity 2.15 Evaluating change

Before the RfC is closed an evaluation (a review) of the whole cycle is performed and lessons learned are stored into the CC-DB.

3.4.1.4 Activity 2.16 Closing change

Finalization of an individual change process.

3.4.2 Essential documents

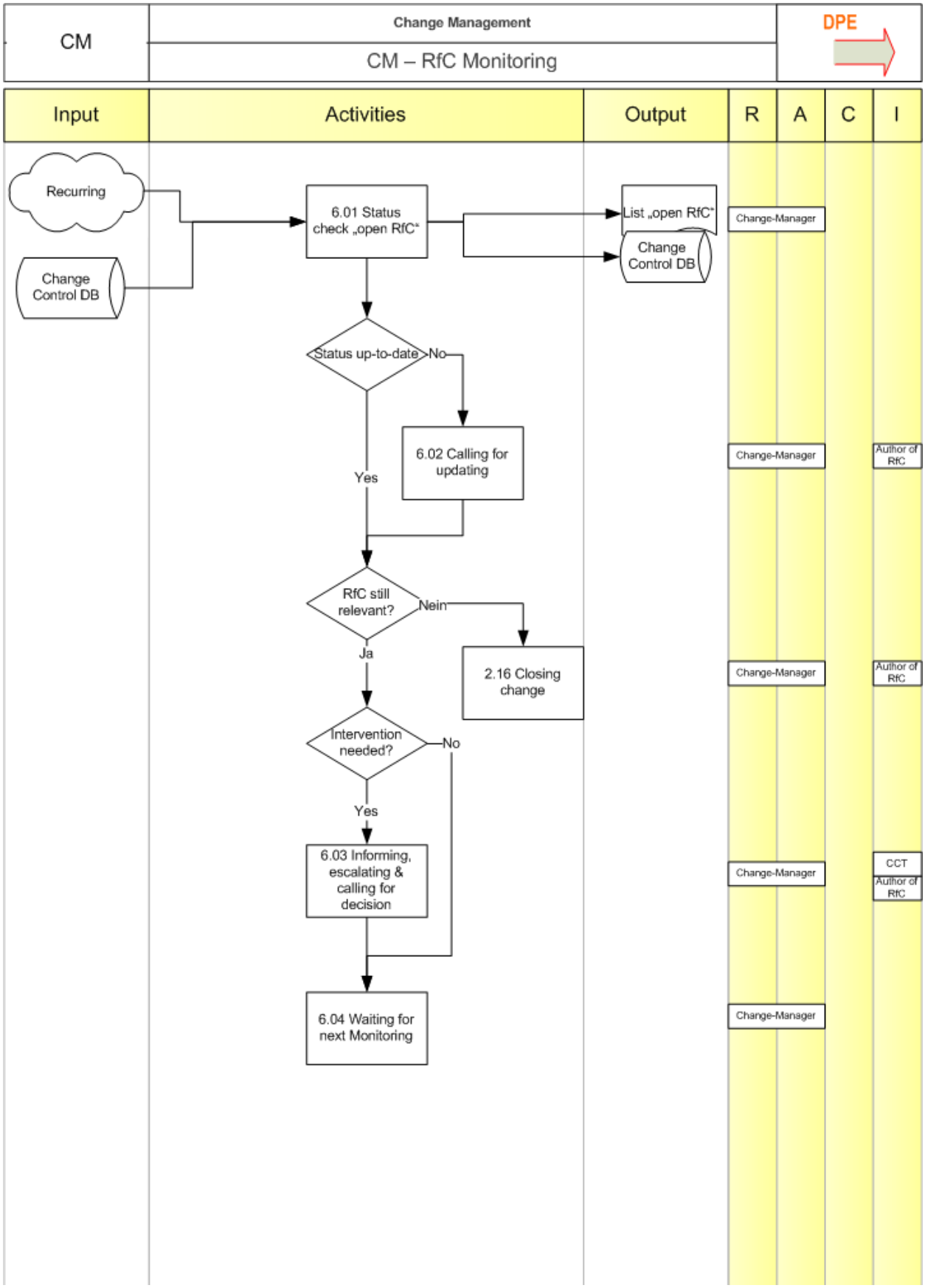
3.4.2.1 Proof of change - Template

Template supporting activity 2.13.

3.4.2.2 Change Review - Template

Template that supports the change review, activity 2.15.

3.5 CM – RfC Monitoring



3.5.1 Core activities

3.5.1.1 Activity 6.01 Status check „open RfC

Periodically the responsible role – the Change Manager – has to check the validity of the status of the RfC stored into the CC-DB. This is performed supported by a report on the open RfC (see 3.5.2.1).

3.5.1.2 Activity 6.02 Calling for updating

In case the status of a specific RfC is not clear or not up to date, the Change Manager asks for an update.

3.5.1.3 Activity 6.03 Informing, escalating & calling for decision

If there is any showstopper within the process flow of an individual RfC, the Change Manager triggers the solution of this situation.

3.5.1.4 Activity 6.04 Waiting for next monitoring

The periodicity of the RfC Monitoring is defined according to the dynamic of the target area.

3.5.2 Essential documents

3.5.2.1 List „open RfC

A simple list of all the RfC with the status “open”; provided by the CC-DB.

3.5.3 Roles

3.5.3.1 Change-Manager

Responsible role to perform the RfC Monitoring.

4 SW-Development (SWD)

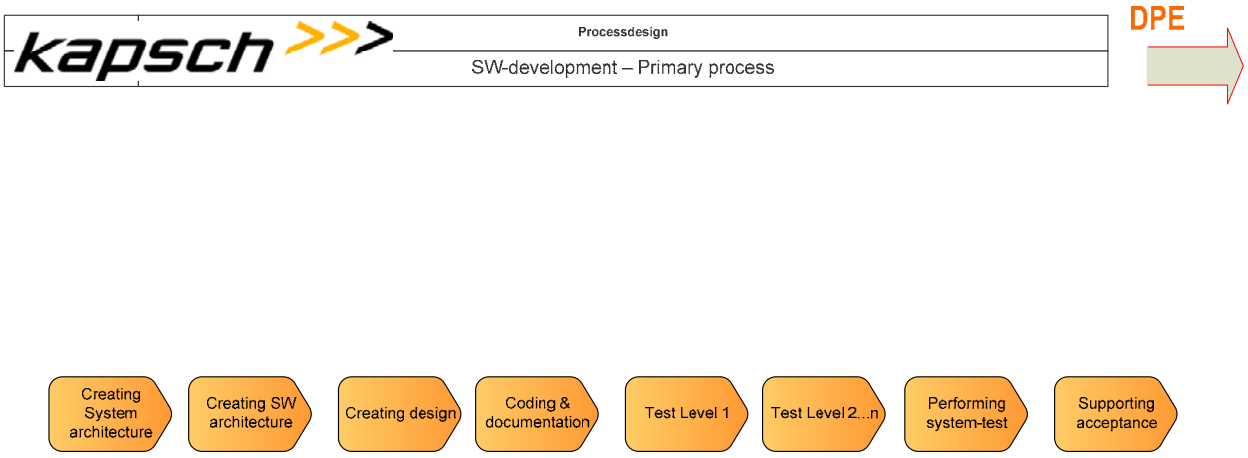
Software-Development (SWD) is the process which organizes the activities concerning the creation of software or more general IT-systems. The main goal of the process is to deliver software which is appropriate to fulfill the needs defined by the process of requirements engineering.

Hence, the process is strongly linked to Requirements Engineering, Test Management and Project Management.

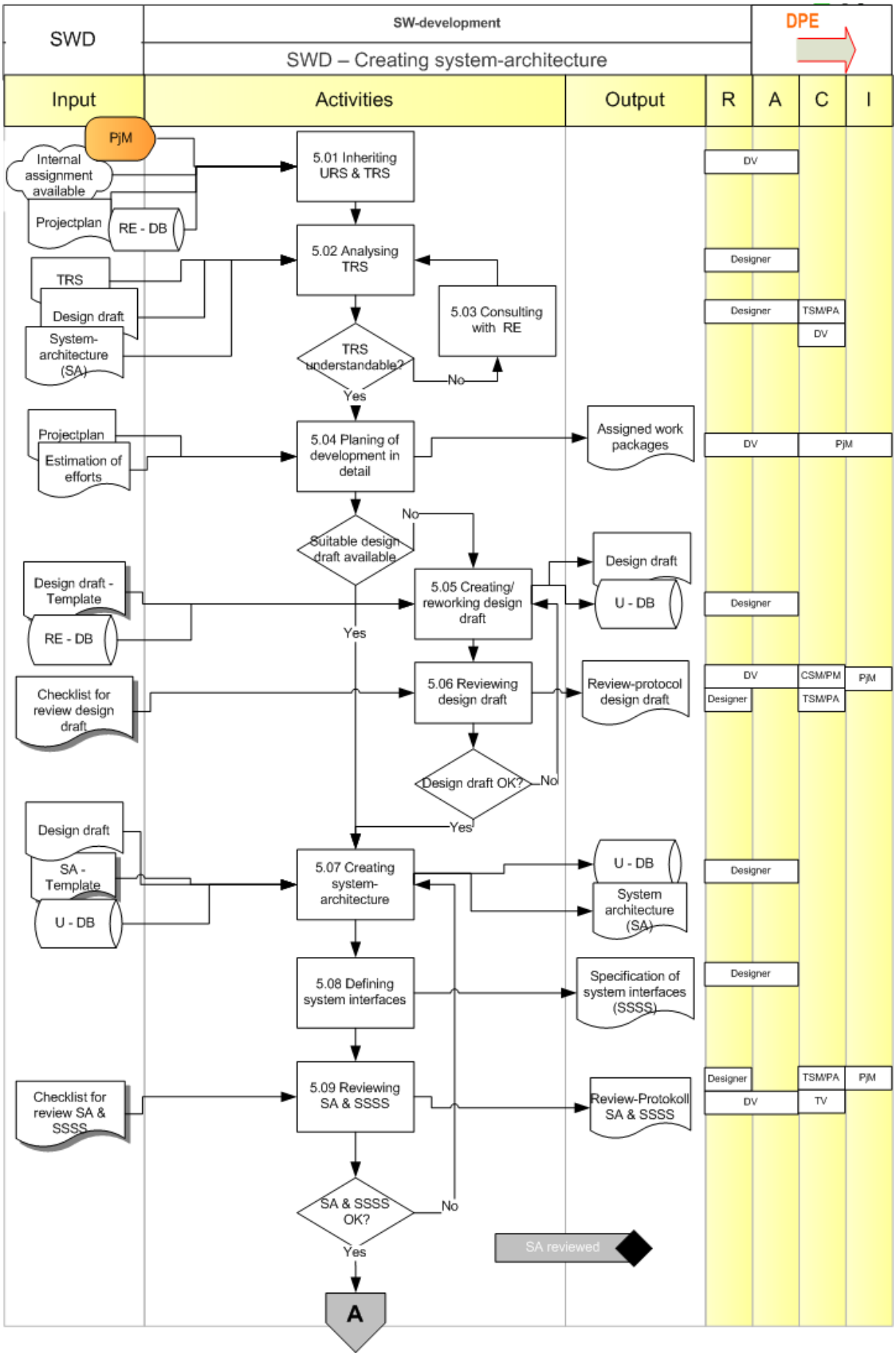
The execution of the process is supported by a tool with the working title U-DB (abbreviation of its name “Umsetzungsdatenbank” in German).

4.1 Primary process

On this level SW-Development is decomposed into eight linearly connected processes: Creating System-Architecture, Creating SW-Architecture, Creating design, Coding & documentation, Test Level 1, Test Level 2…n, Performing system-test and Supporting acceptance.



4.2 SWD – Creating system architecture



4.2.1 Core activities

4.2.1.1 Activity 5.01 Inheriting URS & TRS

The essential inputs for the process of SW-Development are taken over.

4.2.1.2 Activity 5.02 Analyzing TRS

At entrance to the process of SW-Development the Designer analyses the TRS.

4.2.1.3 Activity 5.03 Consulting with RE

In case there are parts of the TRS which the Designer does not understand, this activity is performed.

4.2.1.4 Activity 5.04 Planning of development in detail

This activity delivers a more detailed plan of the various tasks of the SW-Development.

4.2.1.5 Activity 5.05 Creating / reworking design draft

Depending on the quality of the design draft (created by the process of RE), it is reworked or created newly.

4.2.1.6 Activity 5.06 Reviewing design draft

For validation purposes the design draft is reviewed.

4.2.1.7 Activity 5.07 Creating system-architecture

Based on a specific template the system architecture (SA) is created.

4.2.1.8 Activity 5.08 Defining system-interfaces

Supplementing the SA the system-interfaces are defined (SSSS, for the German word “System-Schnittstellen Spezifikation”)

4.2.2 Essential documents

4.2.2.1 Design draft

See the respective template.

4.2.2.2 Project plan

Work-product of the process of Project management

4.2.2.3 Review-protocol design draft

Protocol which is created on the basis of a template.

4.2.2.4 Review-protocol SA & SSSS

Protocol which is created on the basis of a template.

4.2.2.5 System architecture (SA)

Template for the creation of the SA

4.2.2.6 Specification of system interfaces (SSSS)

Template for the creation of the SSSS

4.2.3 Roles

4.2.3.1 Designer

Role which is responsible for most pasts of this secondary process.

4.2.3.2 DV

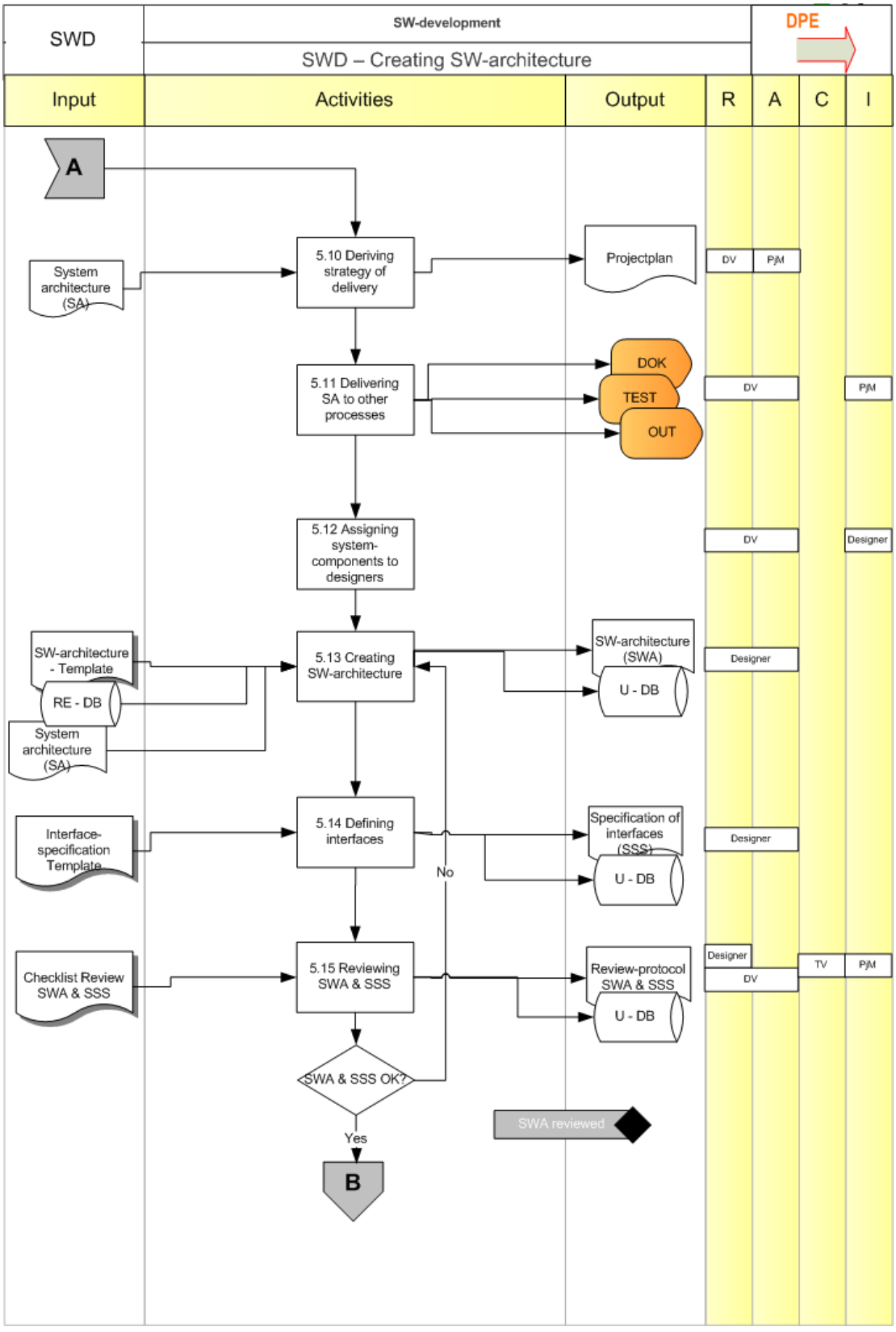
Head of Design (DV for the German “Design-Verantwortlicher”)

4.2.4 Tools

4.2.4.1 U – DB

Application which supports the whole process of SW-Development.

4.3 SWD – Creating SW architecture



4.3.1 Core activities

4.3.1.1 Activity 5.10 Deriving strategy of delivery

Based on the SA the strategy of delivery is derived (if applicable).

4.3.1.2 Activity 5.11 Delivering SA to other processes

Depending on the specific organization of the actual project, the SA is delivered to one or more other processes.

4.3.1.3 Activity 5.12 Assigning system-components to designers

For further detailing the SA is distributed to the Designers

4.3.1.4 Activity 5.13 Creating SW-architecture

As the most detailed input to the coding the software-architecture (SWA) is created using a specific template

4.3.1.5 Activity 5.14 Defining interfaces

As a supplement to the SWA the interfaces are defined using a specific template

4.3.1.6 Activity 5.15 Reviewing SWA & SSS

Documented in a specific protocol SWA and SSS are reviewed

4.3.2 Essential documents

4.3.2.1 SW-architecture (SWA)

Work product of activity 5.13

4.3.2.2 Specification of interfaces (SSS)

Work product of activity 5.14

4.3.2.3 Checklist Review SWA & SSS

Guideline for the execution of activity 5.15.

4.3.3 Processes

4.3.3.1 DOK

Process of documentation

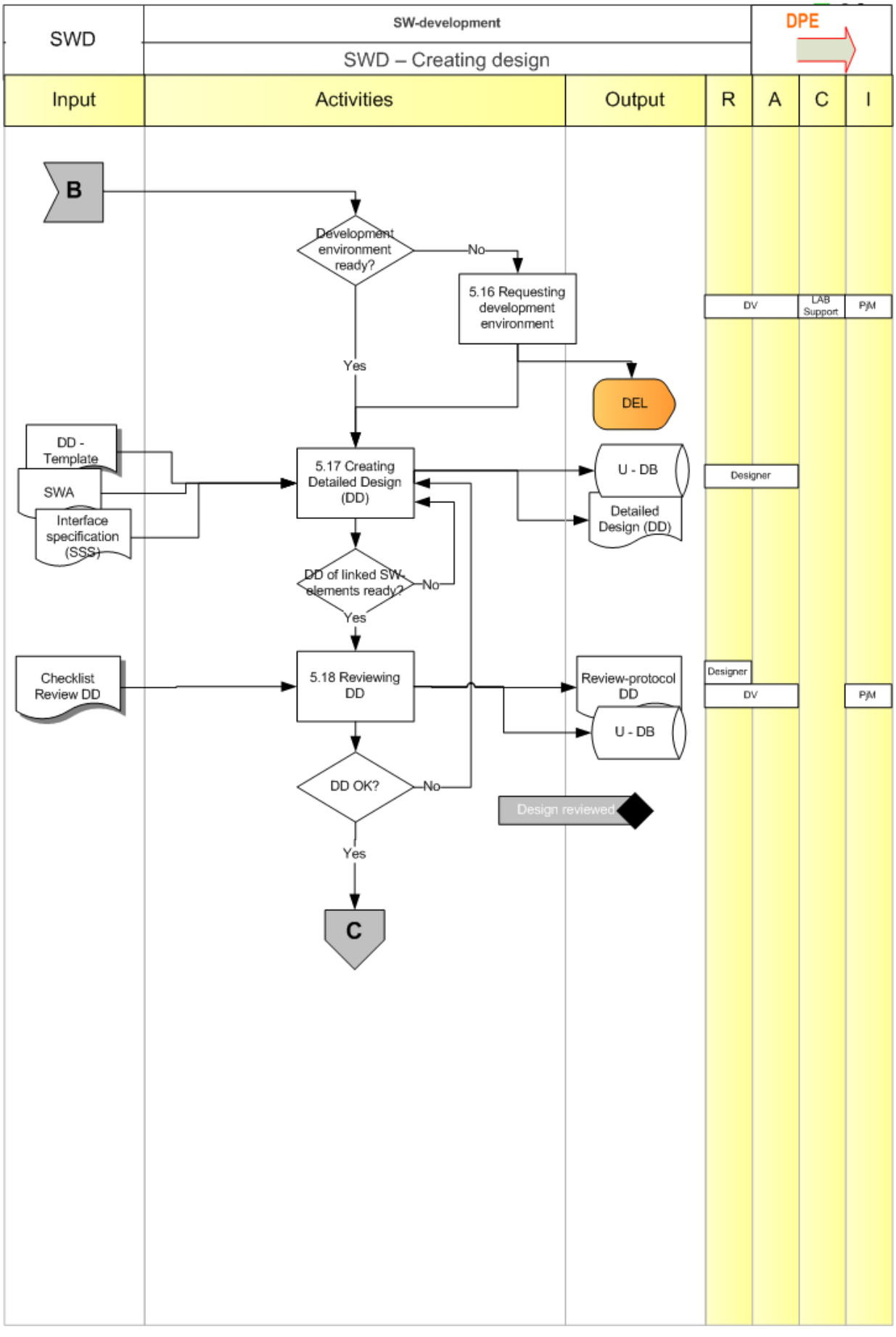
4.3.4 OUT

Process of outtasking (see 6).

4.3.5 TEST

Process of test management (see 5).

4.4 SWD – Creating design



4.4.1 Core activities

4.4.1.1 Activity 5.16 Requesting development environment

The goal of this activity is to be provided with an environment for development and unit-test by the process of DEL.

4.4.1.2 Activity 5.17 Creating Detailed Design (DD)

The deliverable of this activity is the most specific description of the to be created system – the detailed design (DD)

4.4.1.3 Activity 5.18 Reviewing DD

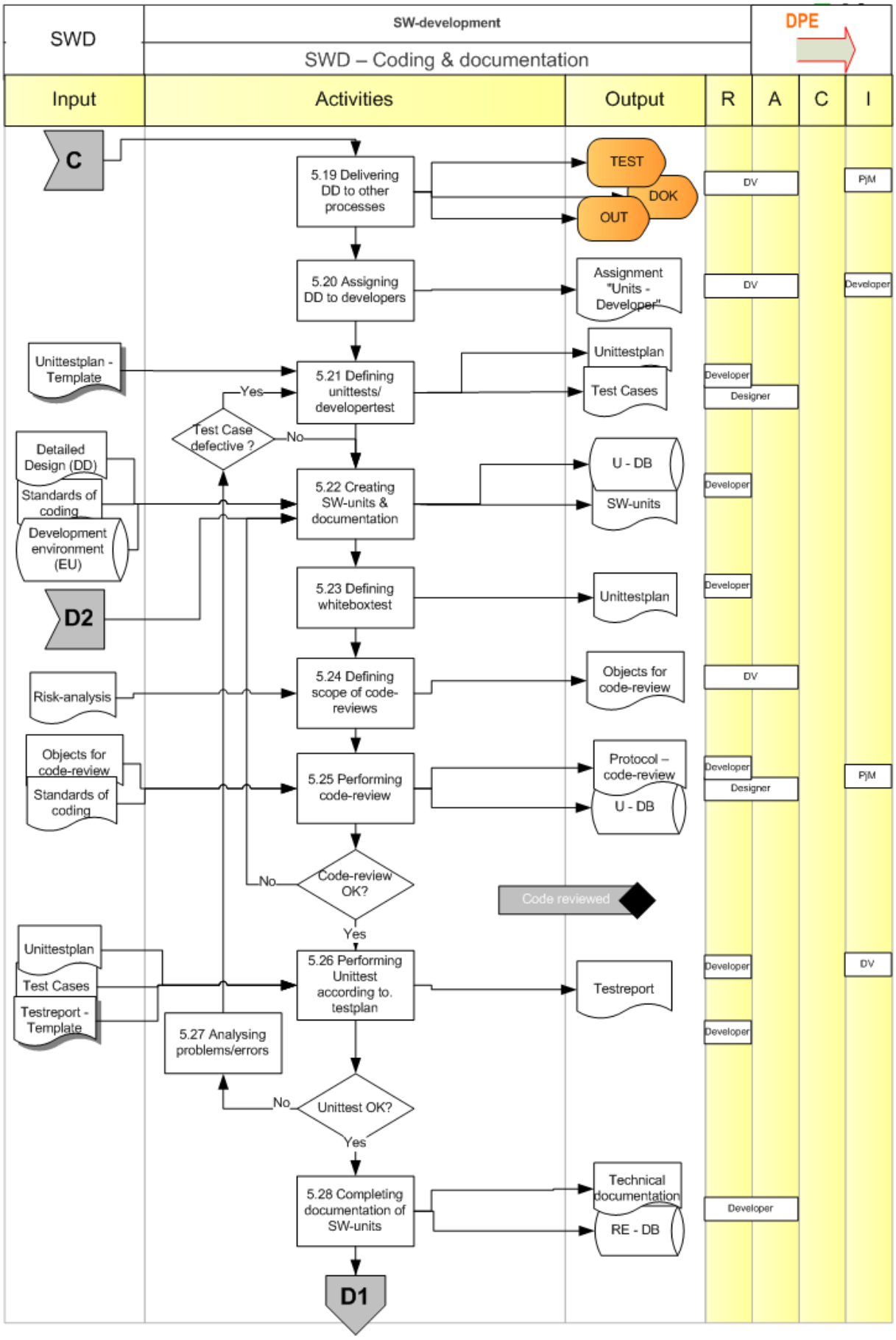
For validation purposes the DD is reviewed by the Head of design. This activity is only performed when the DDs of all relevant SW-elements (objects) are ready. The review is based on a specific checklist and is documented using a protocol.

4.4.2 Essential documents

4.4.2.1 DD – Template

Template for the creation of the detailed design

4.5 SWD – Coding & documentation



4.5.1 Core activities

4.5.1.1 Activity 5.19 Delivering DD to other processes

After the DD has passed its review it is delivered to TEST, OUT and DOK

4.5.1.2 Activity 5.20 Assigning DD to developers

The head of design assigns the DD unambiguously to the developers of the project.

4.5.1.3 Activity 5.21 Defining Unittests/Developertests

Based on the overall test plan the individual developers define their plan for the unit test. This addresses the segmentation in white- and black box-tests.

4.5.1.4 Activity 5.22 Creating SW-units & documentation

According to given priority and coding-standards the individual units are created

4.5.1.5 Activity 5.23 Defining white-box-test

After having created the code, the developers plan the white-box-test.

4.5.1.6 Activity 5.24 Defining scope of code-review

The head of design defines which units should undergo a code-review

4.5.1.7 Activity 5.25 Performing code-review

The code-reviews are performed and documented using a protocol-template

4.5.1.8 Activity 5.26 Performing unit test according to testplan

Unit tests are performed by the developers according to the specific testplans. The unit tests are documented using the testreport-template.

4.5.1.9 Activity 5.27 Analyzing problems/errors

Problems/Errors are analyzed if they occur. Documentation of these problems/errors is not needed in a specific style.

4.5.1.10 Activity 5.28 Completing documentation of SW-units

After successful finalization of the unit test, the technical documentation of the SW-units is completed

4.5.2 Essential documents

4.5.2.1 Standards of coding

These standards could be available and then they are binding

4.5.2.2 Detailed Design (DD)

Work product of activity 5.17

4.5.2.3 Unittestplan

Work product of activity 5.12; created on the basis of a template

4.5.2.4 Technical documentation

Documentation which help to understand the code as designed by the developer

4.5.2.5 Document: Testreport

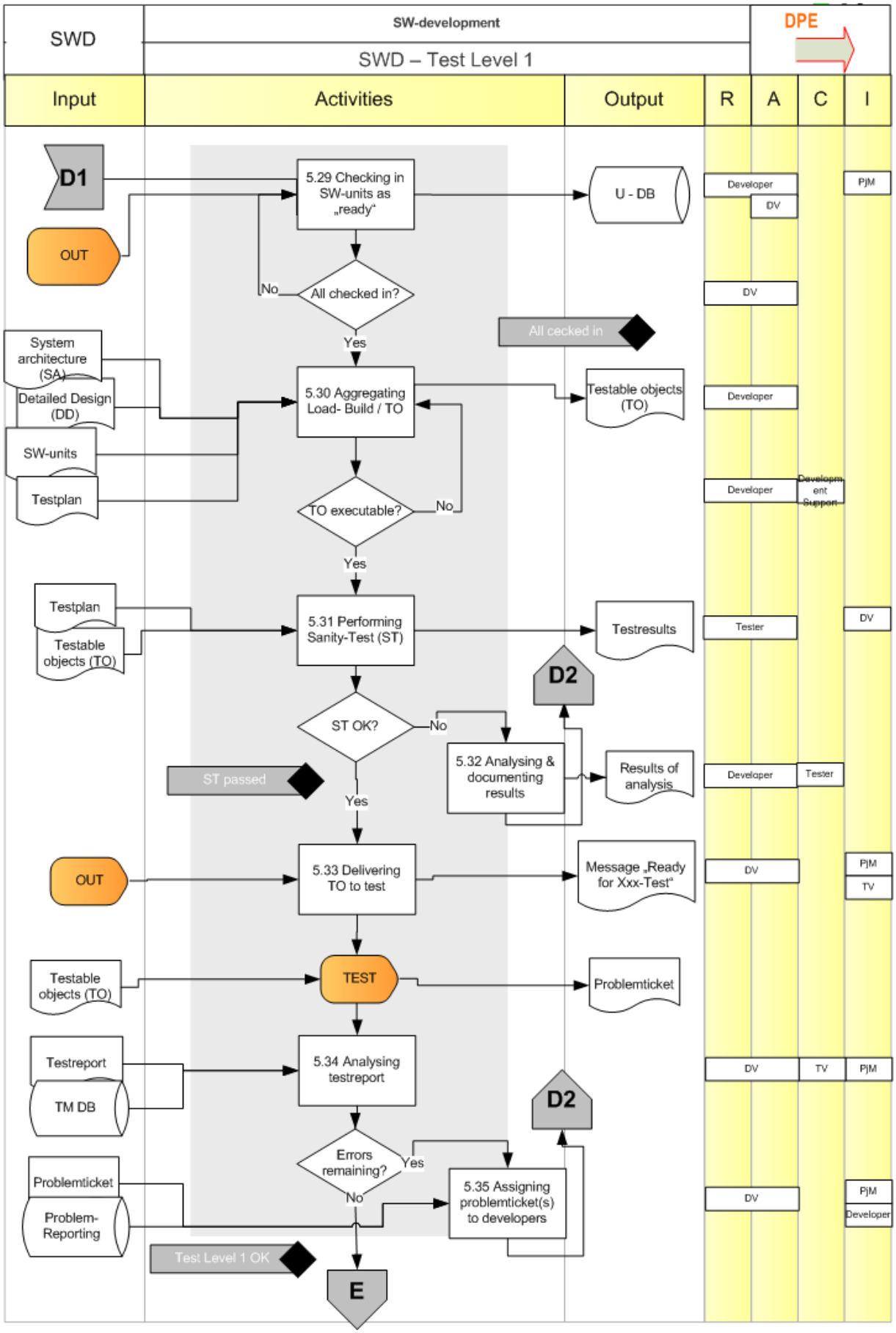
Summarizing documentation of a individual test-run; created on the basis of a template

4.5.3 Roles

4.5.3.1 Developer

Role which is in charge of the creation of the software

4.6 SWD – Test Level 1



4.6.1 Core activities

4.6.1.1 Activity 5.29 Checking in SW-units as „ready”

After having passed the unit tests the individual software units are checked in into the version-control-tool (represented by the U-DB).

4.6.1.2 Activity 5.30 Aggregating Load-Build / TO

After all relevant units have been checked in, the testable objects (TO) as defined in the SA are created

4.6.1.3 Activity 5.31 Performing Sanity-Test (ST)

As a quality gate after the aggregation of the TO the sanity test is performed

4.6.1.4 Activity 5.32 Analyzing & documenting results

Activity which analyses and documents the outcomes of the sanity-test.

4.6.1.5 Activity 5.33 delivering TO to test

After having passed the ST, the testable objects are delivered to the process of test management.

4.6.1.6 Activity 5.34 Analyzing testreport

Test Management returns the testreport which is analyzed in this activity

4.6.1.7 Activity 5.35 Assigning problem tickets to developers

In case errors/problems were detected (by test management) these are assigned to the respective developers. This is supported by a tool with the working title “problem reporting”.

4.6.2 Essential documents

4.6.2.1 Results of analysis

Documentation of the problem detected during the ST and the solution to it

4.6.2.2 Message „Ready for Xxx-Test“

Formal notification which also includes the definition of the type of test (e.g. re-test due to certain reasons)

4.6.2.3 Document: Problem ticket

Formal documentation of a problem which has been detected by test management

4.6.2.4 Document: SW-unit

Smallest individual element which is handled by source-control

4.6.2.5 Document: Testable objects (TO)

Aggregations on various levels which could be executed and hence tested; Load Build

4.6.2.6 Document: Test results

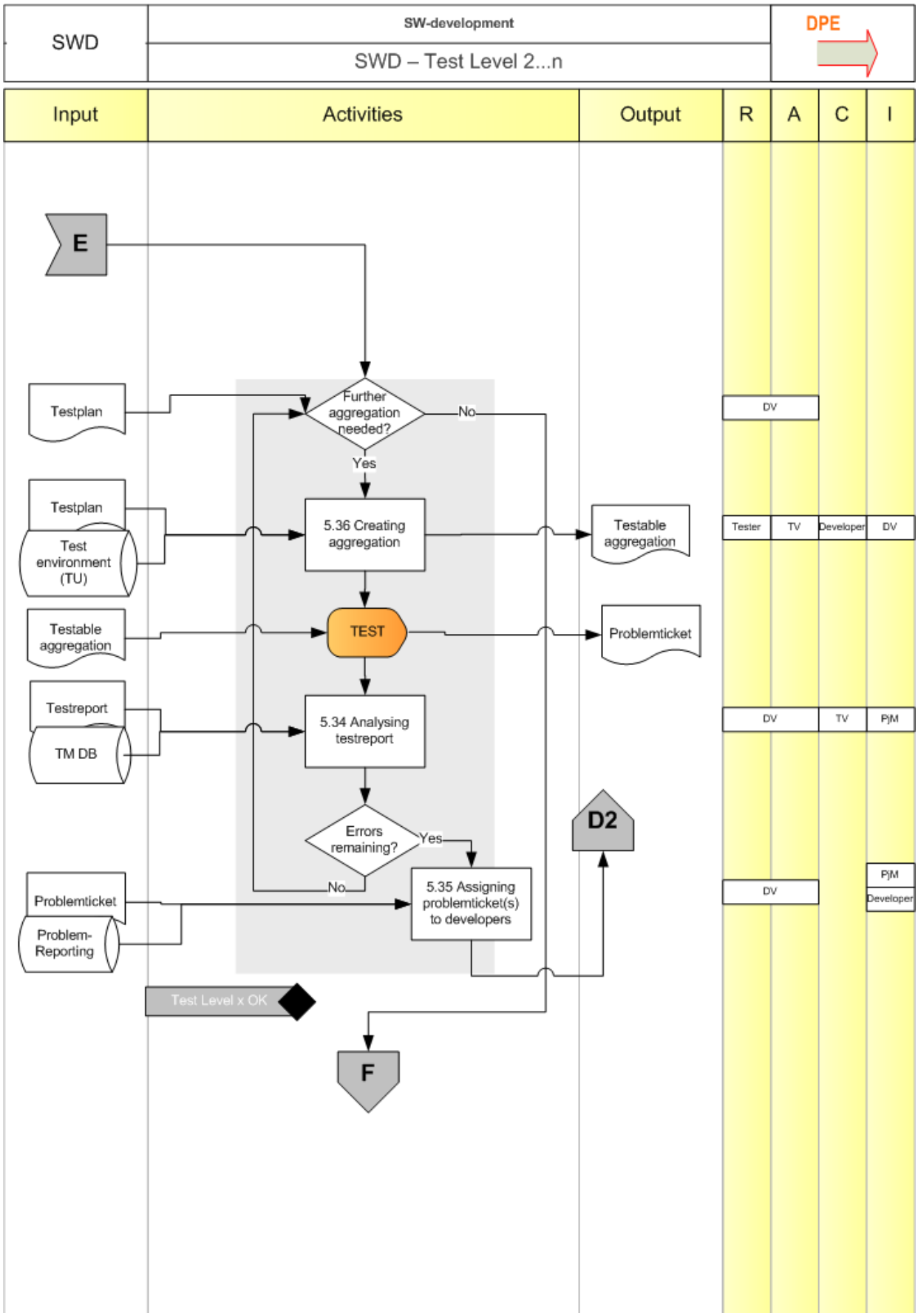
Outcomes of the sanity test

4.6.3 Tools

4.6.3.1 Problem-Reporting

Application which supports the procedures of test management

4.7 SWD – Test Level 2...n

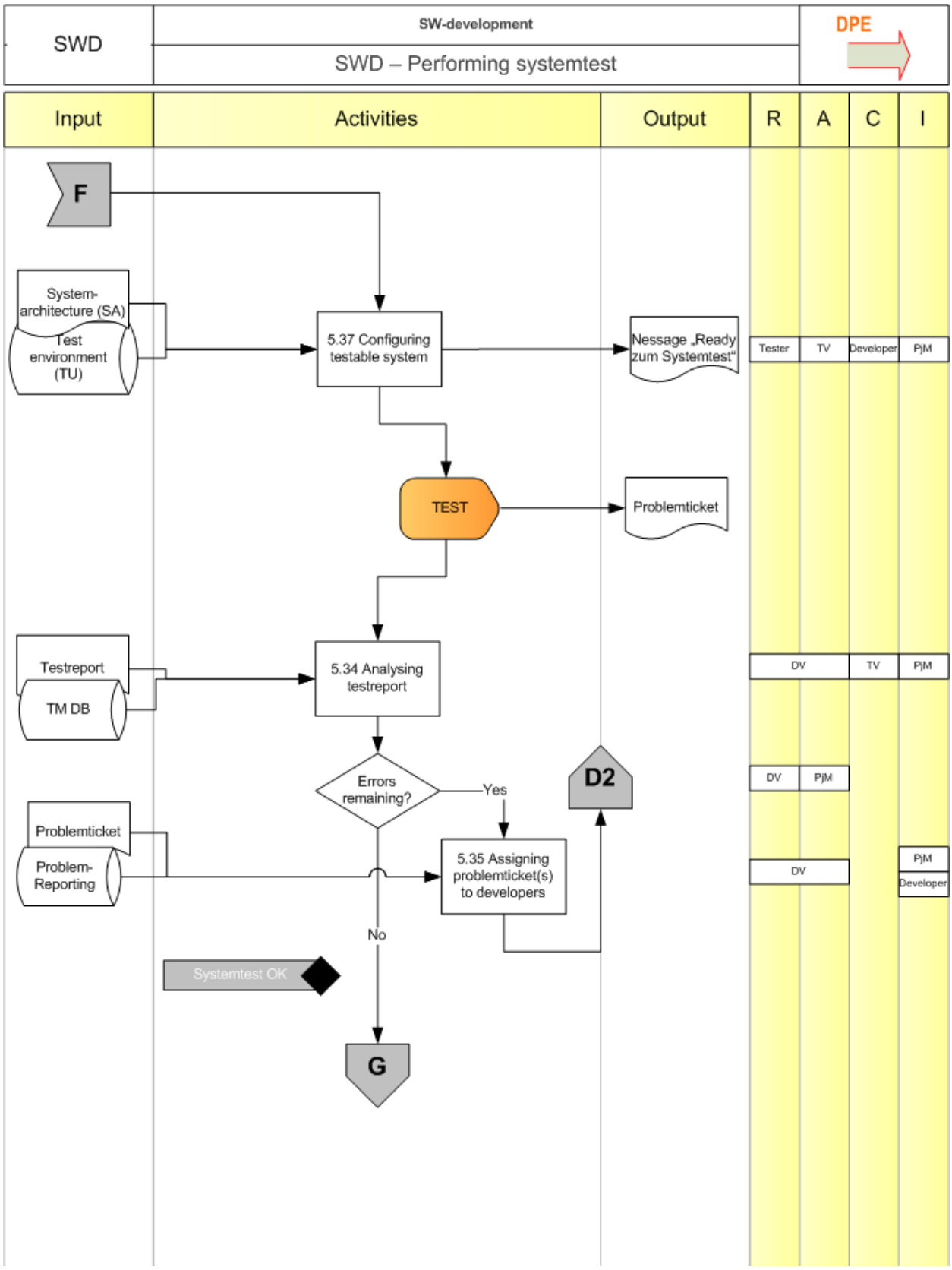


4.7.1 Core activities

4.7.1.1 Activity 5.36 Creating aggregation

In case further aggregations are necessary, testable aggregations (e.g. testable objects on a higher level) are created. This is done in an iterative way up to one test-run prior system-test (see 4.8).

4.8 SWD – Performing system-test



4.8.1 Core activities

4.8.1.1 Activity 5.37 Configuring testable system

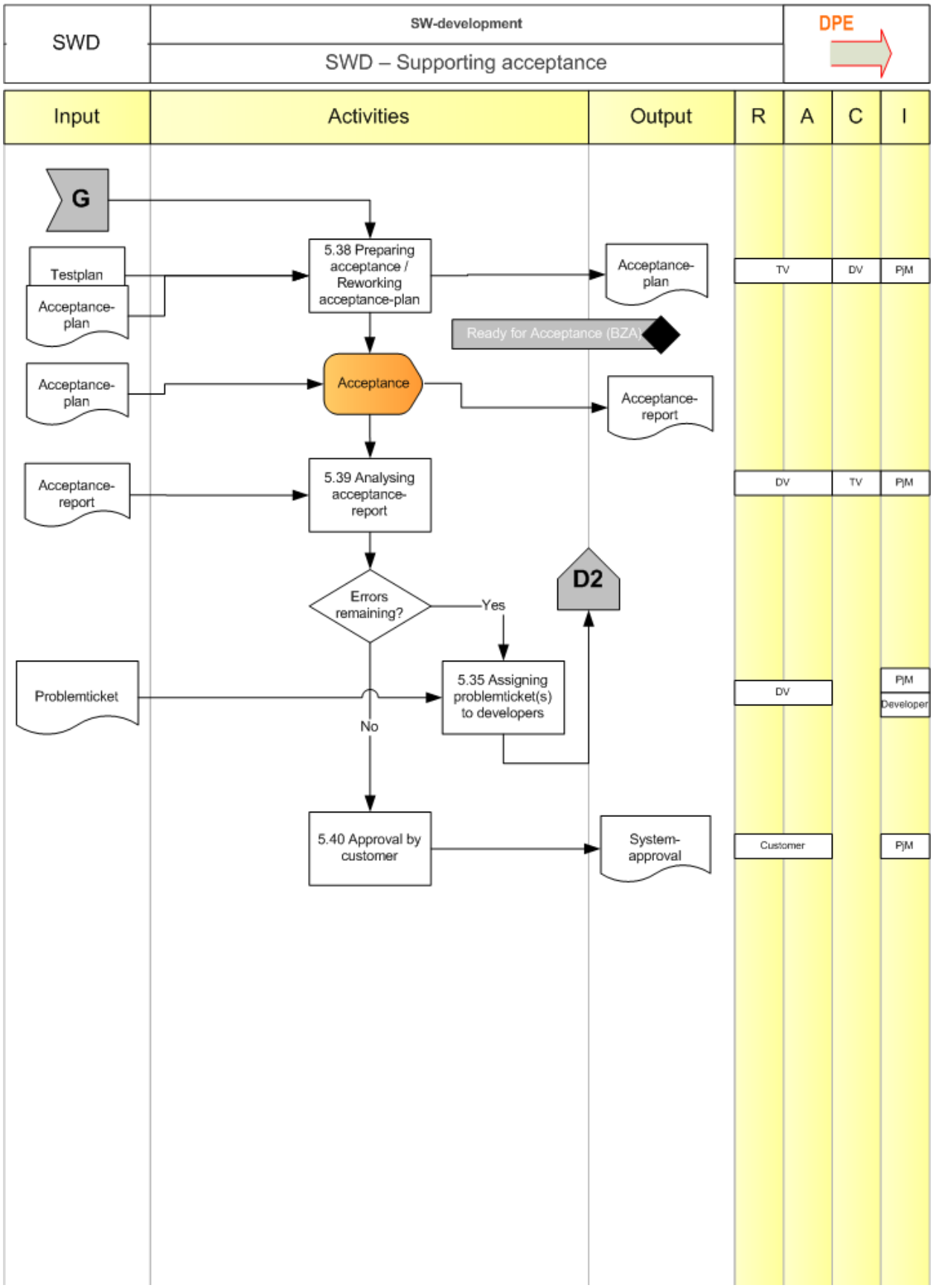
Creating the highest aggregation during the individual project

4.8.2 Essential documents

4.8.2.1 Message „Ready for system test“

Formal notification to the process of test management

4.9 SWD – Supporting acceptance



4.9.1 Core activities

4.9.1.1 Activity 5.38 Preparing acceptance / reworking acceptance-plan

The goal of this activity is to check and occasionally rework the acceptance-plan

4.9.1.2 Activity 5.39 Analyzing acceptance-report

The process of Acceptance returns a specific report which is to be analyzed in this activity

4.9.1.3 Activity 5.40 Approval by customer

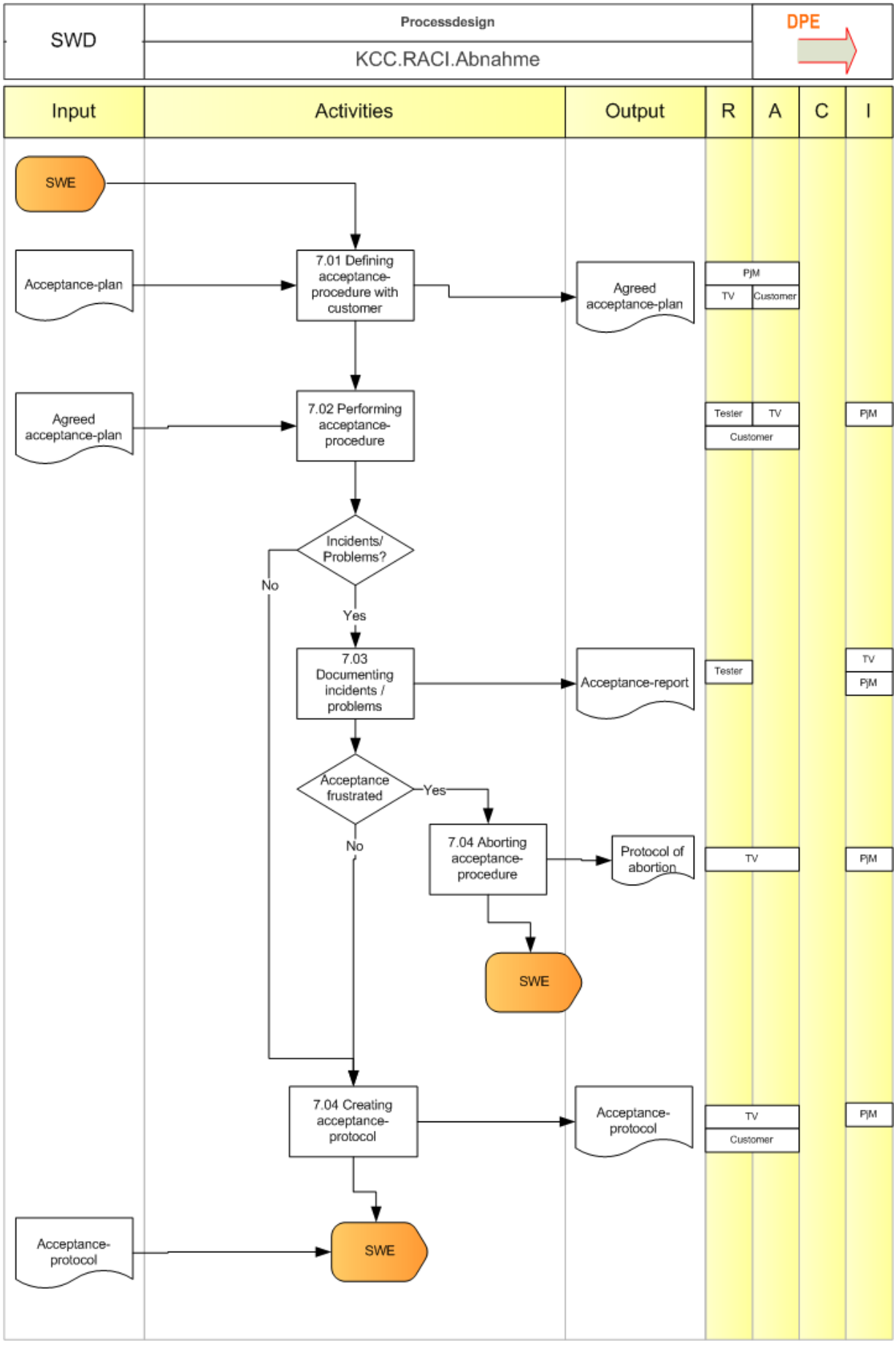
This activity is performed by the customer and represents the formal step

4.9.2 Processes

4.9.2.1 Acceptance

See 4.10

4.10 Acceptance



4.10.1 Core activities

4.10.1.1 Activity 7.01 Defining acceptance-procedure with customer

Based on the acceptance-plan which has been created unilateral, a bilateral procedure (“agreed acceptance plan”) is defined

4.10.1.2 Activity 7.02 Performing acceptance-procedure

According to the agreed plan the acceptance is carried out

4.10.1.3 Activity 7.03 Documenting incidents/problems

Incidents and problems are documented according to the agreed procedures

4.10.1.4 Activity 7.04 Creating acceptance-protocol

In case acceptance has been reached a formal protocol is created

4.10.1.5 Activity 7.04 Aborting acceptance-procedure

In case acceptance has been failed a formal protocol is created

4.10.2 Essential documents

4.10.2.1 Protocol of abortion

Documentation of the abortion of acceptance

4.10.2.2 Document: (Agreed) acceptance-plan

Documentation of the acceptance-procedure which both sides agreed upon

4.10.2.3 Document: Acceptance-report

Documentation of the acceptance-test

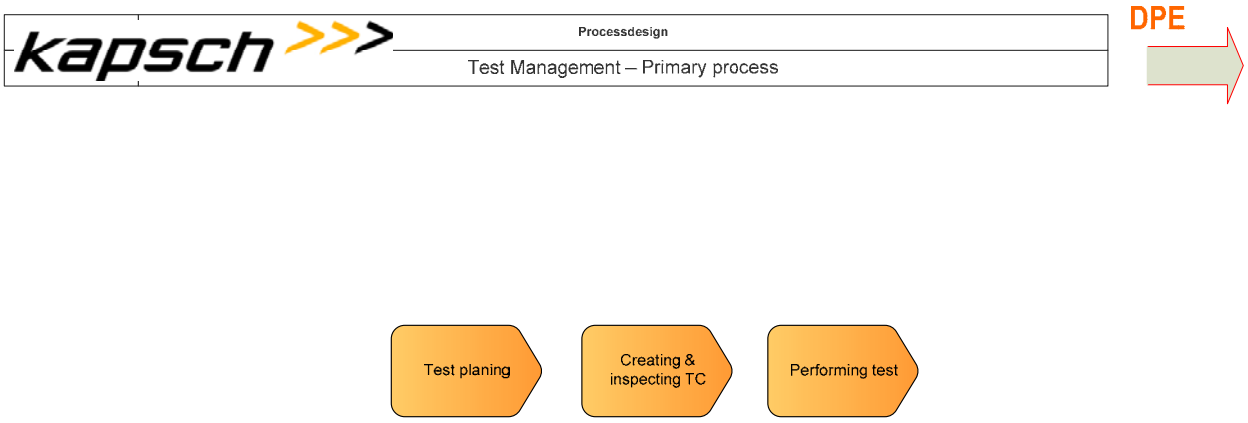
5 Test Management (TM)

Test Management is the process that organizes al activities concerning testing. The main goal of the process is to deliver software with assured level of quality. The process is triggered basically by the project management and strongly linked to the process of SW-Development.

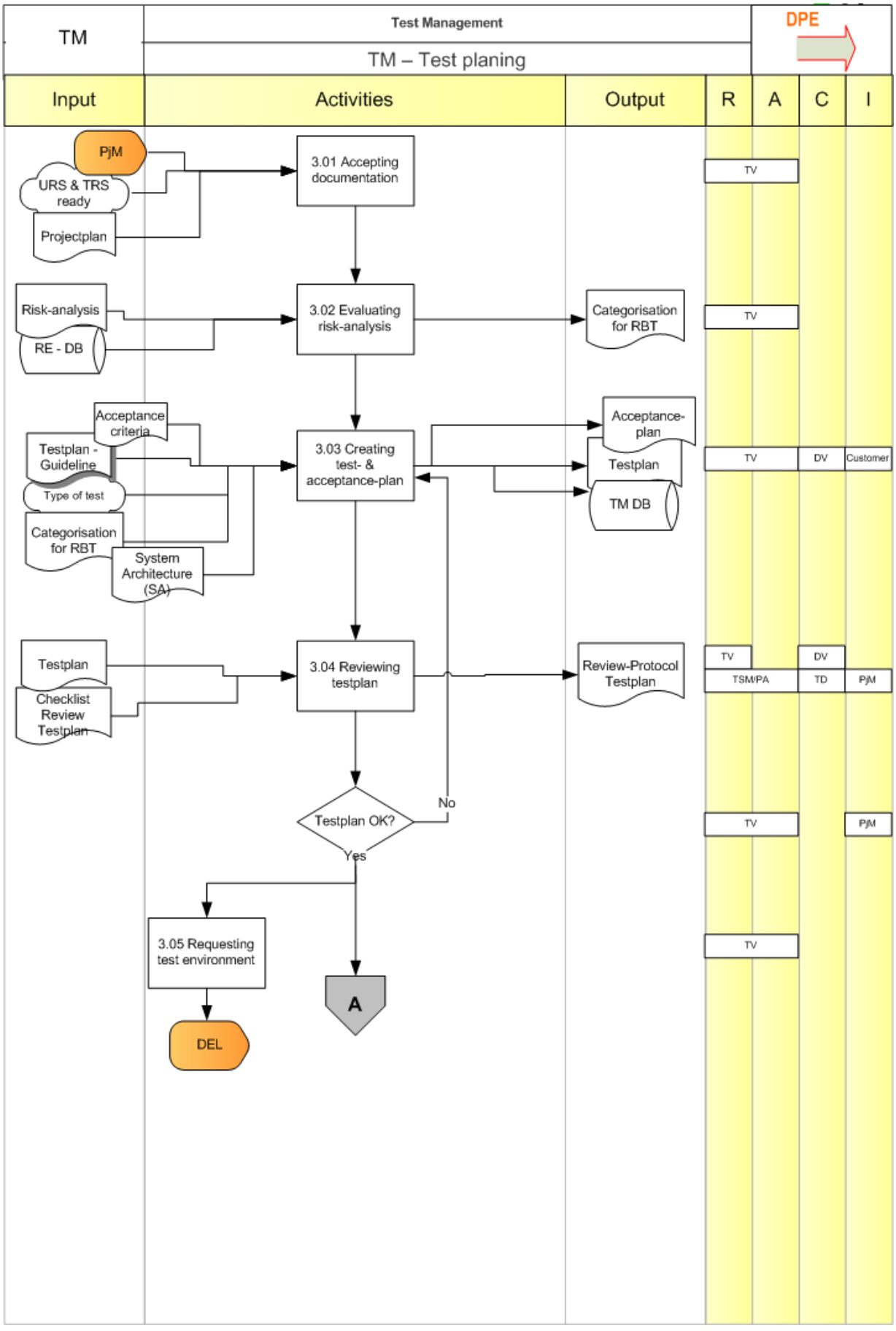
The execution of the process is supported by a tool with the working title TM-DB (i.e. test management database).

5.1 Primary process

On this level the process of test management is decomposed into three linearly connected processes: Test planning, Creating & inspecting TC and performing test.



5.2 TM – Test planning



5.2.1 Core activities

5.2.1.1 Activity 3.01 Accepting documentation

After URS and TRS are ready (i.e. passed the review) they are handed over to the test management process.

5.2.1.2 Activity 3.02 Evaluating risk-analysis

The actual risk-analysis is analyzed and used for the categorization for the risk based testing (RBT).

5.2.1.3 Activity 3.03 Creating test- & acceptance-plan

Based on several sources the test- and acceptance-plan is created and stored into the TM-DB.

5.2.1.4 Activity 3.04 Reviewing testplan

Using a specific checklist (see 5.2.2.2) the testplan is reviewed by a team. In case some roles collapse into one person, the project manager assigns the review team.

5.2.1.5 Activity 3.05 Requesting test environment

According to the definitions of the testplan a services-request is delivered to the Del process. The test environment is designed according to the type of test (integration, regression, system, acceptance, emergency patch)

5.2.2 Essential documents

5.2.2.1 Acceptance-plan

Section of the testplan.

5.2.2.2 Checklist Review Testplan

Guideline for the review of the testplan.

5.2.2.3 Project plan

Deliverable of the project management process.

5.2.2.4 Review-protocol Testplan

Formal documentation of the review of the testplan.

5.2.2.5 Risk-analysis

Base document deriving from the RE process.

5.2.2.6 System Architecture (SA

Base document for the creation of the testplan.

5.2.2.7 Testplan

Central document of the test management process. Defines the test procedure on the different levels (if applicable) and stages. Determines the conditions for test stop & test breakup, the demands on the test environment and the strategy regarding regression.

5.2.3 Processes

5.2.3.1 DEL

Process which creates test-environment or organizes (e.g. a reference-system on the customer-side) creation of it.

5.2.4 Roles

5.2.4.1 Customer

Representatives of the customer

5.2.4.2 TD

Role which is responsible for the design/creation of the test cases.

5.2.4.3 TV

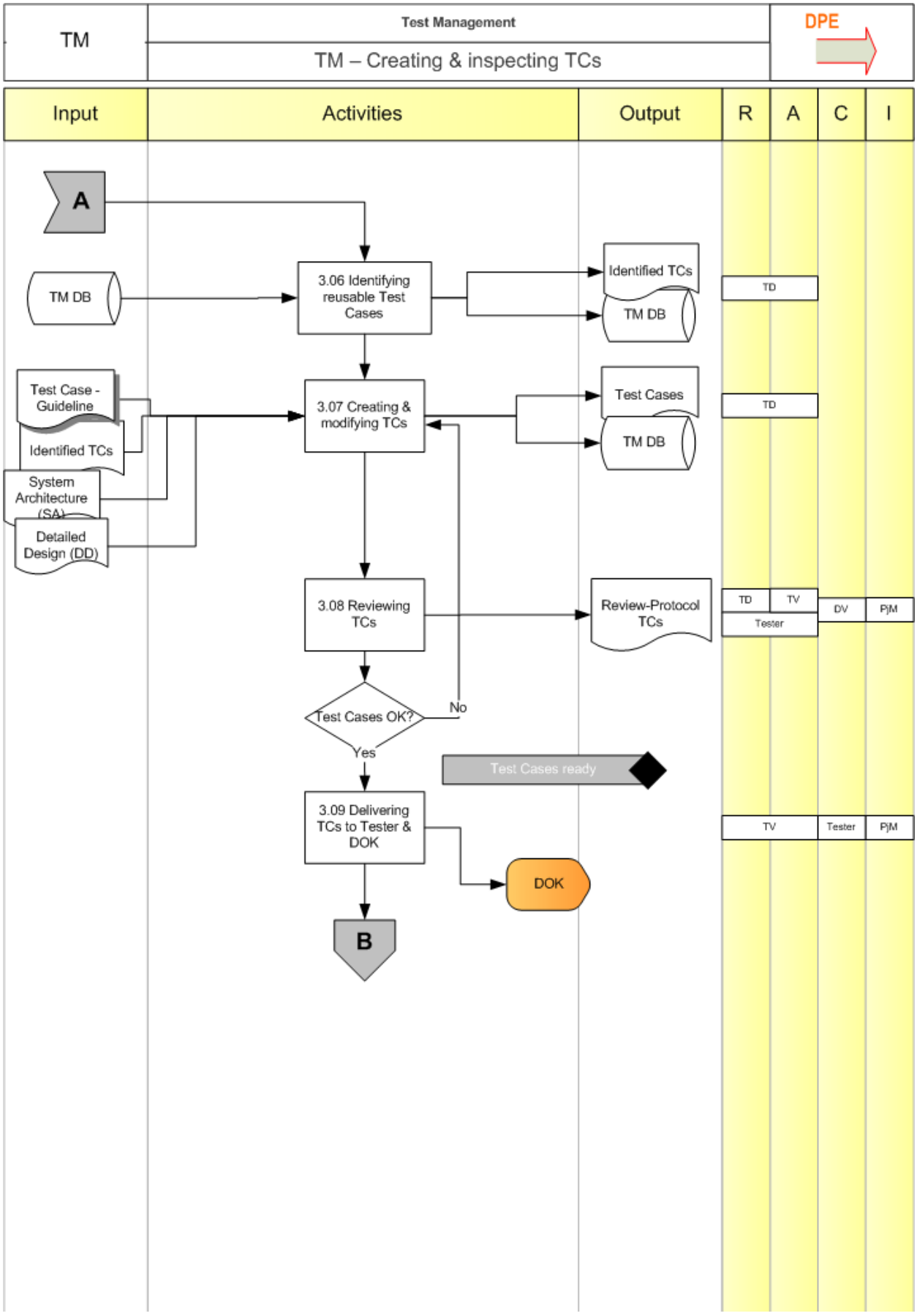
Head of test – responsible role for planning and executing tests.

5.2.5 Tools

5.2.5.1 TM DB

Tool/application which supports the whole process of test management.

5.3 TM – Creating & inspecting TC



5.3.1 Core activities

5.3.1.1 Activity 3.06 Identifying reusable Test Cases

The goal of this activity is to find already defined test cases (TC) in the TM-DB

5.3.1.2 Activity 3.07 Creating & modifying TCs

This activity delivers TCs which are appropriate for the actual test process. TCs are created on the basis of a specific template (see 5.3.2.3).

5.3.1.3 Activity 3.08 Reviewing TCs

A review of the test cases is performed before they are delivered to the process of test management

5.3.2 Essential documents

5.3.2.1 Identified TCs

List of TC which could be re-used

5.3.2.2 Review-Protocol TCs

Formal documentation of the outcomes of the review

5.3.2.3 Test Case

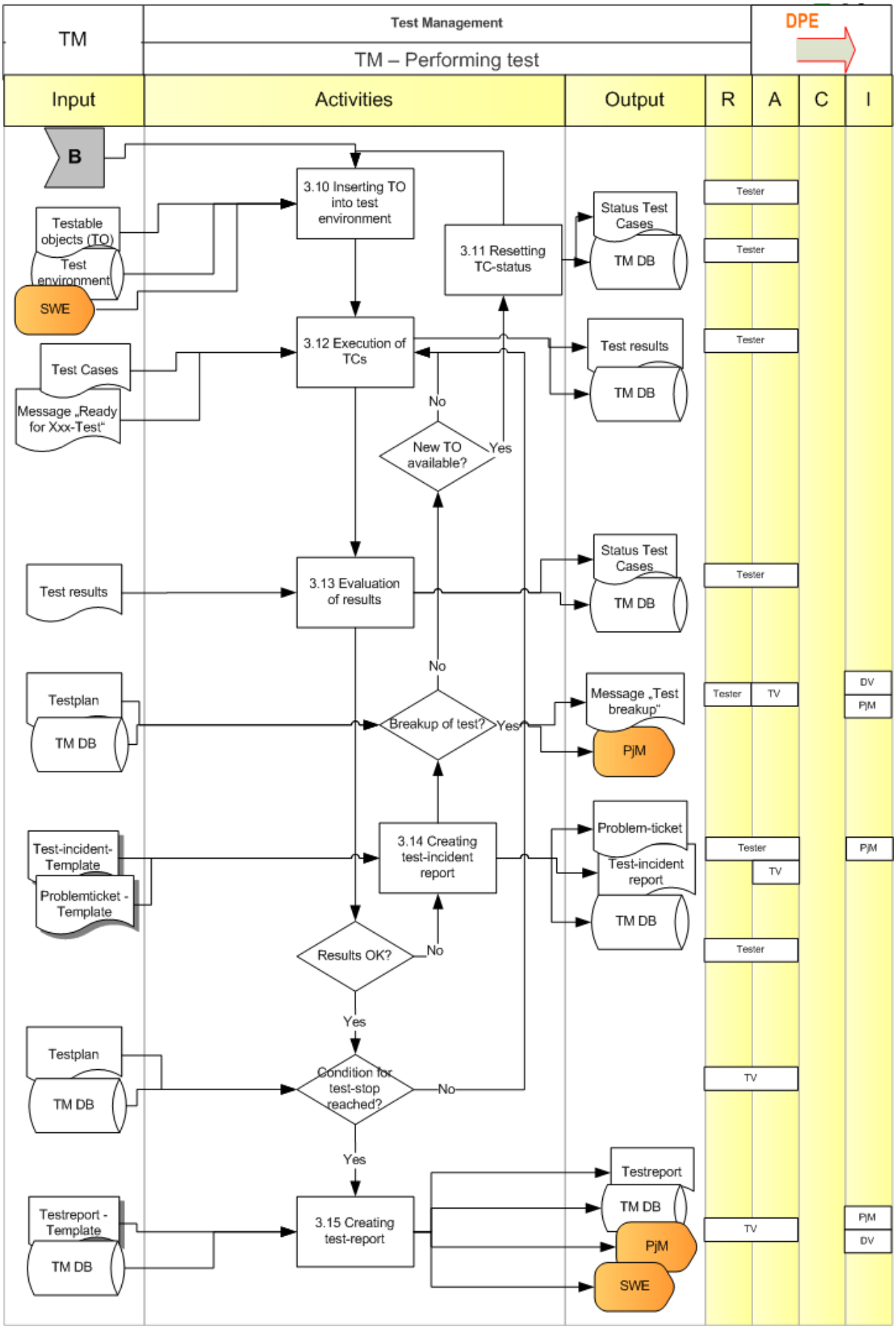
Document describing single tasks of a test, created on the basis of a specific template

5.3.3 Roles

5.3.3.1 Tester

Role responsible for reviewing executing TCs

5.4 TM – Performing test



5.4.1 Core activities

5.4.1.1 Activity 3.10 Inserting TO onto test environment

The task of this activity is to install the TOs on the test environment

5.4.1.2 Activity 3.12 Execution of TCs

The test cases are executed as defined

5.4.1.3 Activity 3.11 Resetting TC-status

In case it is necessary (e.g. a re-run of TCs) the status of individual TCs is reset. Possible status-changes are as follows: initial – reviewed – failed – passed – reopened.

5.4.1.4 Activity 3.13 Evaluating results

The results of the test-run is stored into the TM-DB

5.4.1.5 Activity 3.14 Creating test-incident report

Incidents that occur are documented using a specific template

5.4.1.6 Activity 3.15 Creating test-report

Summing up all results of the test-run

5.4.2 Essential documents

5.4.2.1 Message „Test breakup”

Formal notification documenting the breakup of the test-run. This occurs when a significant number of TCs could not be executed

5.4.2.2 Test-incident report

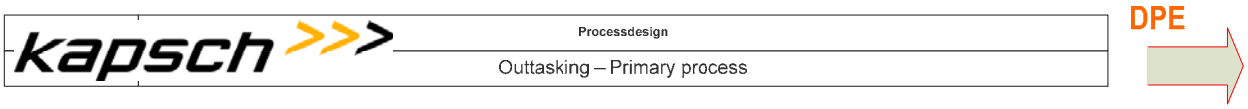
Document which is created on the basis of a specific template documenting unexpected results of a test case

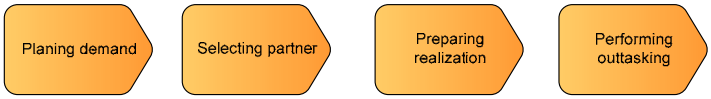
6 Outtasking (OUT)

Outtasking is the process that organizes all activities concerning the cooperation with partner during the execution of a project. The main goal of the process is to manage the distributed fulfillment of an assignment. The process is strongly connected to Project management and SW-Development.

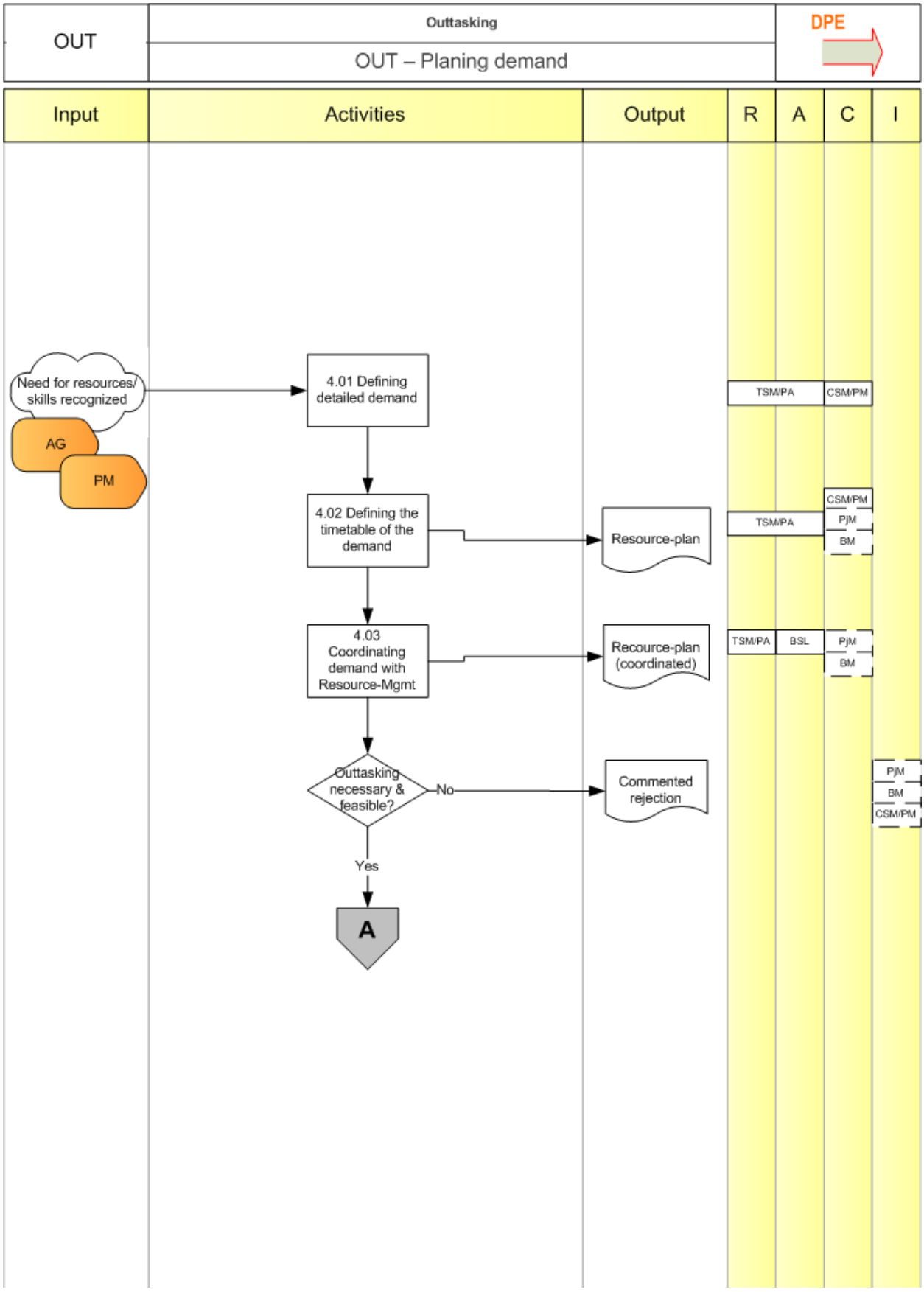
6.1 Primary process

On this level the process of Outtasking is decomposed into four linearly connected secondary processes: Planning demand, Selecting partner, Preparing realization and Performing outtasking.





6.2 OUT – Planning demand



6.2.1 Core activities

6.2.1.1 Activity 4.01 Defining detailed demand

The task of this activity is to create a detailed plan of demand after the need for resources/skills has been recognized

6.2.1.2 Activity 4.02 defining the timetable of the demand

The deliverable of this activity is a resource-plan

6.2.1.3 Activity 4.03 Coordinating demand with Resource-Mgmt

The goal of this activity is to agree on the plan and its impacts

6.2.2 Essential documents

6.2.2.1 Resource-plan (coordinated)

Work product of activity 4.03

6.2.2.2 Commented rejection

Documentation of the decision not to outtask certain activities

6.2.2.3 Resource-plan

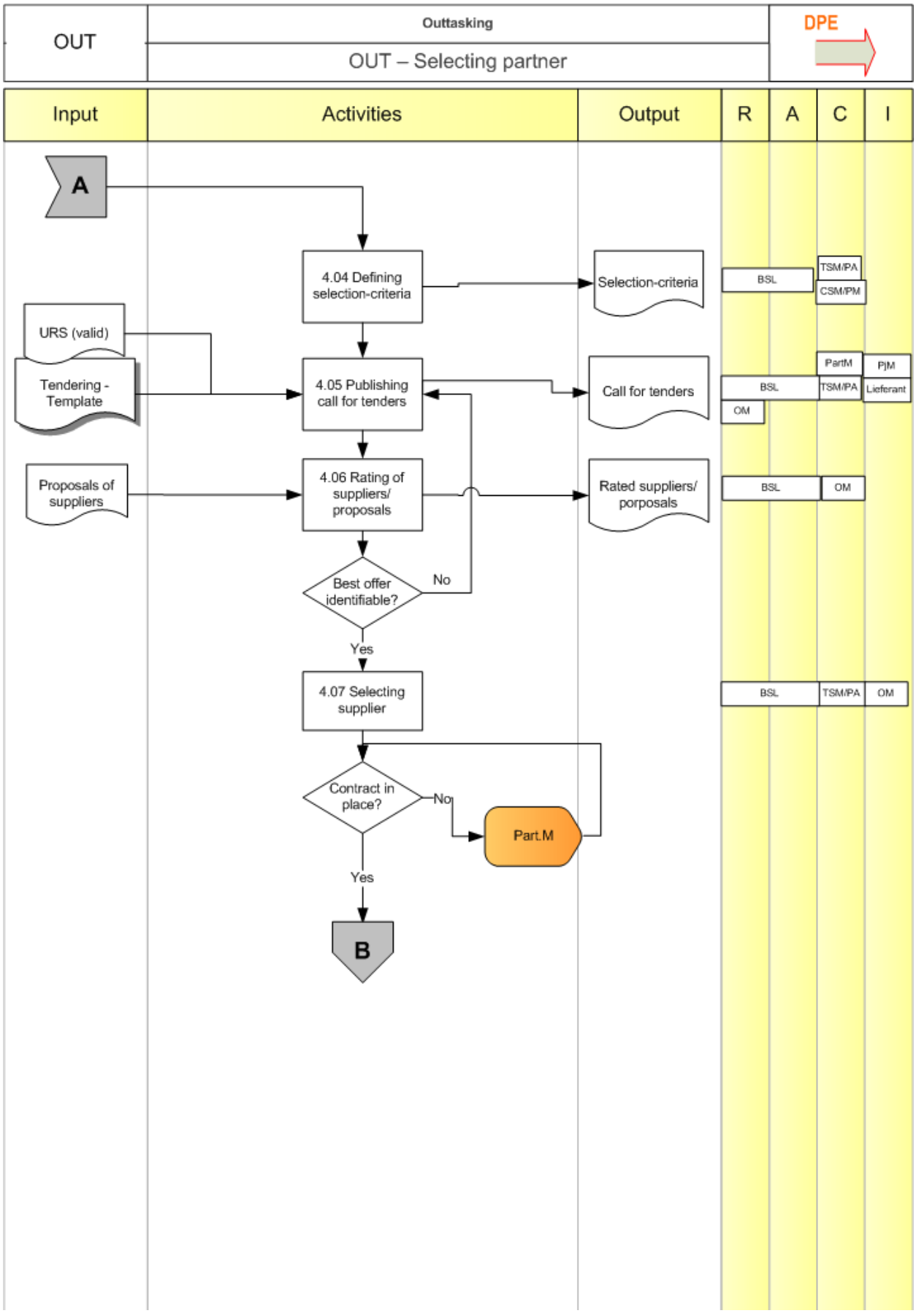
Documentation of needed skills, timelines and costs

6.2.3 Roles

6.2.3.1 BSL

Manager, responsible for a certain business unit

6.3 OUT – Selecting partner



6.3.1 Core activities

6.3.1.1 Activity 4.04 Defining selection-criteria

Creation of a scheme for selection of the appropriate supplier

6.3.1.2 Activity 4.05 Publishing call for tenders

Formal step of communicating the tender documents

6.3.1.3 Activity 4.06 Rating of suppliers/proposals

Evaluating the proposals and the suppliers

6.3.1.4 Activity 4.07 Selecting supplier

Deciding on the best offer

6.3.2 Essential documents

6.3.2.1 Call for tenders

Base document of the tendering procedure

6.3.2.2 Proposals of suppliers

Answer of the tenderer

6.3.3 Processes

6.3.3.1 Part.M

Process of Partner Management, see respective definition on intranet

6.3.4 Roles

6.3.4.1 Supplier

Tenderer, “Lieferant” in German

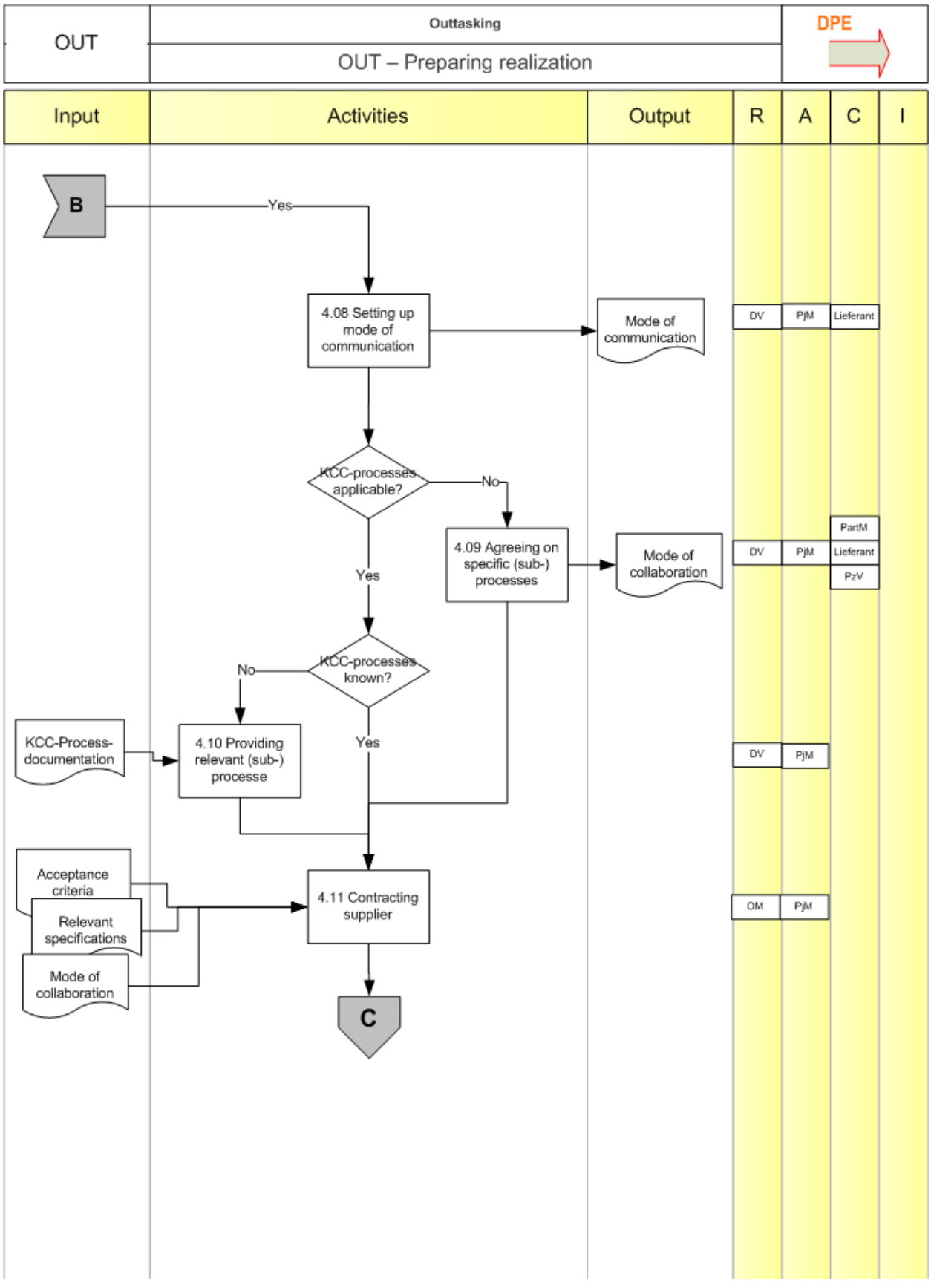
6.3.4.2 OM

Order Manager, responsible role for ordering

6.3.4.3 PartM

Partner Manager, responsible role in the process of Partner Management

6.4 OUT – Preparing realization



6.4.1 Core activities

6.4.1.1 Activity 4.08 Setting up mode of communication

The goal of this activity is to agree upon an appropriate mode of communication during the actual project

6.4.1.2 Activity 4.09 Agreeing on specific (sub-)processes

In case the KCC processes are applicable it is agreed on a specific subset of them

6.4.1.3 Activity 4.10 Providing relevant (sub-)processes

If the KCC processes are not known yet they are provided to the supplier

6.4.1.4 Activity 4.11 Contracting supplier

On the basis of various defining documents the contract with the supplier is closed

6.4.2 Essential documents

6.4.2.1 KCC-Process documentation

The document at hand

6.4.2.2 Mode of communication

Agreement on the principles of communication between KCC and supplier

6.4.2.3 Mode of collaboration

Agreement on the used processes

6.4.2.4 Relevant specifications

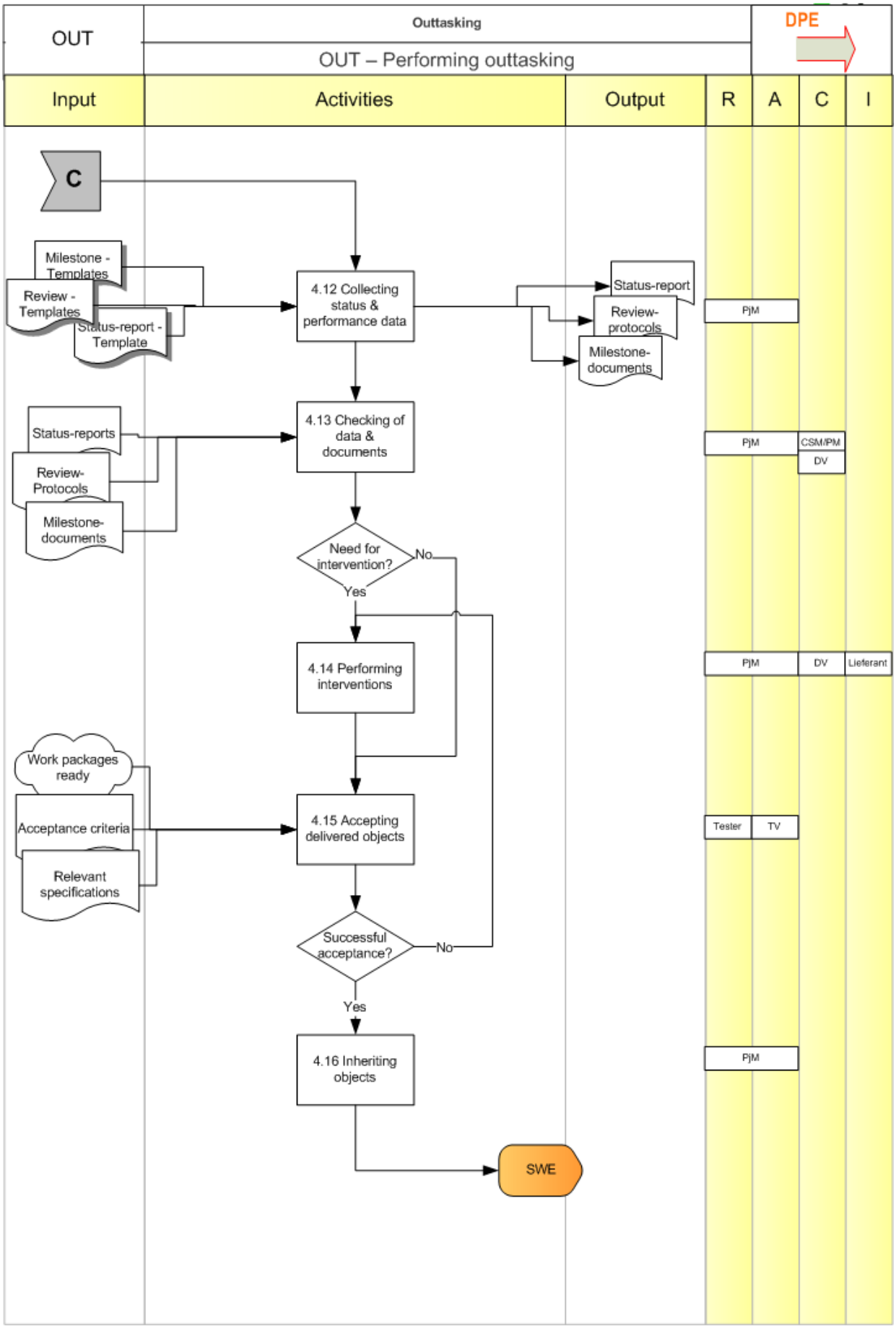
Base documents for the work the supplier has to fulfill

6.4.3 Roles

6.4.3.1 PzV

Process Manager, responsible role regarding the relevant process(es)

6.5 OUT – Performing outtasking



6.5.1 Core activities

6.5.1.1 Activity 4.12 Collecting status & performance data

The documentation of various kinds (progress, compliance) is collected by the project-manager

6.5.1.2 Activity 4.13 Checking data & documents

Performing an assessment of the documents delivered by the supplier

6.5.1.3 Activity 4.14 Performing interventions

According to the assessment an intervention is performed

6.5.1.4 Activity 4.15 Accepting delivered objects

If the deliverable matches the relevant criteria they are accepted

6.5.1.5 Activity 4.16 Inheriting objects

The objects are transferred as defined in the respective contract (e.g. specifications, designs, software, lessons learned)

6.5.2 Essential documents

6.5.2.1 Milestone-documents

Documentation of the project management process

6.5.2.2 Review-protocols

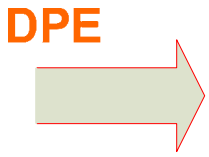
Documentation of the process of SW-development

6.5.2.3 Document: Status-reports

Documentation of the project management process

7 Dictionary of terms

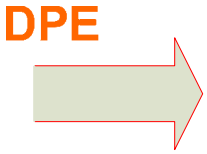
Term / Expression	Synonym in German	Explanation
Acceptance	ABNAHME	
Acceptance frustrated?	Abnahmeverhindern?	Acceptance not successful
Acceptance-criteria	Abnahmekriterien	
Acceptance-plan	Abnahmeplan	
Acceptance-procedure	Abnahmeprozedur	Specific activities which are agreed on with the customer and which are carried out during customer-acceptance
Acceptance-protocol	Abnahmeprotokoll	
Acceptance-report	Abnahmereport	Documentation of all incidents or problems which occur during acceptance
Acceptance-plan	Abnahmeplan	Mutual agreed between KCC and customer
Annotation for URS/TRS	Anmerkungen zu URS/TRS	Remarks concerning URS or TRS
Approval by customer	Freigabe durch Kunden	Written documentation of acceptance
Assignment “units – developer”	Zuteilung "Module - Developer"	n:m - relation
Assumptions (ex open issues)	Annahmen (aus OPL)	Remarks which are introduced into URS as special type of requirements
Bid Manager	BM	Role in the process AG



Breakup of test	Testabbruch	Severe cause for breaking up test
BSL	BSL	Head of organizational unit
Call for tender	Ausschreibung	Specific document based on the tendering-template
Categorization for RBT	Kategorisierung für RBT	Base-information for risk-based-testing
Change Control Team	CCT	Decision group for RfC
Change Manager	Change-Manager	Responsible role for RfC-Monitoring
Change Officer	Change-Verantwortlicher	Responsible role for the CM-process
Change-plan	Änderungsplan	Documentation describing the planed change
CMDB	CMDB	Configuration Management Database
Commented rejection	Begründete Ablehnung	State of reason for a denial
Condition for test-stop	Testendebedingung erreicht?	Target to be reached during test
Contractual basis	Vertragliche Grundlagen	Basis for qualification of a RfC
Creating aggregation	Aggregation herstellen	Iterative activity of creating new testable objects
Criteria for validation & verification	Verifikations- & Validierungs-Kriterien	According to V-Model
Customer	Kunde	Organisation and its representatives on the buyer-side
Design draft	Lösungsentwurf	Initial document describing the planned solution
Detail Design	Detailed Design (DD)	Basis for coding

Estimation of efforts	Aufwandsschätzung	Calculated amount of person-days
Evaluation of feasibility	Machbarkeitsbewertung	Assessment of the feasibility
Incidents	Ereignisse	Unexpected events
Interfaces	Schnittstellen	Communcation between sw-objects
Internal assignment	Interne Beauftragung	Documented basis of a project
Mode of collaboration	Mode of collaboration	Agreement on specific processes
Mode of communication	Kommunikationsplan	Base-definition for outtasking
Proof of change	Änderungsnachweis	Documentation that the change has been successful
Protocol of abortion	Abbruchprotokoll	Documentation of abortion of acceptance
RfC	RfC	Request for change
RfC (analysed)	RfC (analysiert)	Different stages of RfC
RfC (approved)	RfC (freigegeben)	
RfC (closed)	Abgeschlossener RfC	
Risk-analysis	Risikoanalyse	Structured evaluation of risk per requirement
S-CCT	S-CCT	Senior CCT
Selection-criteria	Auswahlkriterien	Scheme which is used for finding the best bid
Specification of system interfaces	Systemschnittstellen Spezifikation (SSSS)	
Strategy of delivery	Auslieferungsstrategie	Input to the project-plan based on the system-architecture
SWA	SWA	SW-architecture
SWD	SWE	SW-Development

System architecture	System Architektur (SA)	Initial work-product of SWD
System interfaces	Schnittstellen Spezifikation (SSS)	Documentation of communication between system elements
Tendering – Template	Ausschreibung - Template	Basis for all tenders
Test environment	Testumgebung (TU)	Base infrastructure need for test
Testable aggregation	Testbare Aggregation	Result of aggregating TOs
Testable object	Testbare Objekte (TO)	
Test-incident	Testereignis	
Test-report	Testreport	
Test-result	Testergebnisse	
TRS	TRS	Technical Requirements Specification; includes the requirements of all stakeholders
TRS (valid)	TRS (gültig)	Actual version of the TRS
Type of test	Testart	Integration, regression, system, acceptance, emergency/patch
Unit	SW-Module	Atomic SW-element
Unittestplan	Modultestplan	Testplan for atomic SW-elements
URS	URS	User Requirements Specification; functional requirements; customer-focused
URS (valid)	URS (gültig)	Actual version of the URS
Winning assignments	AG	Process responsible for the pre-assignment-phase
Work package	Arbeitspaket	Defined unit of work


DPE