VDA

Verband der Automobilindustrie

ProSTEP iViP

ECM Recommendation Part 1 (ECR)

Version 2.0, Issued Aug. 2009

Replacements: Version 1.0

VDA 4965 – Part 1

Version 3.0, issued Jan. 2010

Replacements: Version 2.0

SASIG

strategic automotive product data standards industry group

A Joint Publication

SASIG

Automotive Industry Action Group (U.S.), Groupement pour l'Amelioration des Liaisons dans l'Industrie Automobile (France), Japan Automobile Manufacturers Association (Japan) / Japan Auto Parts Industries Association (Japan), Odette Sweden AB (Sweden), and Verband der Automobilindustrie (Germany) are members of the Strategic Automotive product data Standards Industry Group- SASIG. These organisations are cooperating in the creation, distribution and use of joint documents, including PDQ (Product Data Quality), PDM Assembly Data Exchange, XMTD (Exchange and Management of Technical Data), DEV (Digital Engineering Visualization).

SASIG Publications

A SASIG publication reflects a consensus of those substantially concerned with its scope and provisions. A SASIG publication is intended as a guide to aid the manufacturer, the consumer and the general public. The existence of a SASIG publication does not in any respect preclude anyone from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the publication.

Cautionary Notice

SASIG publications are subject to periodic review and users are cautioned to obtain the latest editions.

Maintenance Procedure

Recognizing that this joint publication may not cover all circumstances, SASIG has established a maintenance procedure. Please refer to the Maintenance Request Form at the back of this document to submit a request.

Contact information

Contacts information to SASIG is available on http://www.sasig.com.

Copyright and Trademark Notice

Table of Contents

1 ECR Reference Process Fundamentals 1
1.1 ECR models and messages related to ECM 1
1.2 ECR specific rules 1
2 Communication and the ECR Reference Process 3
2.1 Overview of the modelling concept and the notation used 3
2.2 Overview of the ECR Reference Process 4
2.2.1 Purpose 4
2.2.2 Phases of the ECR Reference Process 4
2.2.3 Normal flow and status 5
2.2.4 Special flow variants and exceptions 7
2.3 ECR organization model and roles 12
2.4 ECR Reference Process 15
2.4.1 Inquiry of ECR phase 15
2.4.2 Creation of ECR phase 18
2.4.3 Technical Analysis of ECR phase 21
2.4.4 Commenting on ECR phase 25
2.4.5 Approval of ECR phase 30
3 ECR Interaction Scenarios and ECR Messages 37
3.1 Description method 38
3.2 ECR Interaction Scenarios 39
3.2.1 ECR Interaction Scenario IS1 – Participant proposal for a change 39
3.2.2 ECR Interaction Scenario IS2 – Participant comments 41
3.2.3 ECR Interaction Scenario IS3 – Participant approval 44
3.2.4 ECR Interaction Scenario IS4 – Participant detailing and comments 47
3.3 ECR Messages 50
3.3.1 Data-oriented ECR Messages 50
3.3.2 Process-oriented ECR Messages 52
3.3.3 Use of ECR Messages within the predefined ECR Interaction Scenarios 60
3.3.4 ECR Message headers and control information 62
4 ECR Data Model 63
4.1 Description method 63
4.2 Usage of the ECM Data Model for the ECR Reference Process 63
4.3 Usage of ECM Data Model objects within ECR Messages 69
Appendix A: ECR Interaction Scenario Protocol Specifications 71
A.1 Purpose & Overview 71
A.2 Cooperation-specific parameters 72
A.3 Protocol Specifications 73

A.3.1 ECR Interaction Scenario 1 Protocol Specification 73
A.3.2 ECR Interaction Scenario 2 Protocol Specification 75
A.3.3 ECR Interaction Scenario 3 Protocol Specification 80
A.3.4 ECR Interaction Scenario 4 Protocol Specification 82

Maintenance Form 87

Figures

Figure 1: Phases of the ECR Reference Process ... 4
Figure 2: Normal flow of ECR Reference Process ... 6
Figure 3: Activity Distribute ECR Update ... 7
Figure 4: Activity Get External ECR ID ... 7
Figure 5: Flow variants of the ECR Reference Process ... 8
Figure 6: Activity Cancel ECR ... 9
Figure 7: Activity Rollback ECR to Analysis ... 11
Figure 8: Activity Rollback ECR to Commenting ... 12
Figure 9: ECR organization model ... 13
Figure 10: Phase overview for Inquiry of ECR ... 16
Figure 11: Normal flow for Inquiry of ECR ... 17
Figure 12: Phase overview for Creation of ECR ... 18
Figure 13: Normal flow for Creation of ECR ... 20
Figure 14: Phase overview for Technical Analysis of ECR ... 21
Figure 15: Normal flow for Technical Analysis of ECR ... 22
Figure 16: Flow variants for Technical Analysis of ECR ... 24
Figure 17: Phase overview for Commenting on ECR ... 25
Figure 18: Normal flow for Commenting on ECR ... 27
Figure 19: Flow variants for Commenting on ECR (1/2) ... 29
Figure 20: Flow variants for Commenting on ECR (2/2) ... 30
Figure 21: Phase overview for Approval of ECR ... 31
Figure 22: Normal flow for Approval of ECR ... 32
Figure 23: Flow variants for Approval of ECR ... 34
Figure 24: Interaction flow intensity in the typical ECR Interaction Scenarios ... 37
Figure 25: Permitted sequences of messages for participant proposal for a change (IS1) .. 40
Figure 26: Messages in IS1 in the context of the ECR Reference Process ... 41
Figure 27: Permitted sequences of messages for participant comments (IS2) ... 42
Figure 28: Messages in IS2 in the context of the ECR Reference Process ... 44
Figure 29: Permitted sequences of messages for participant approval (IS3) ... 45
Figure 30: Messages in IS3 in the context of the ECR Reference Process ... 47
Figure 31: Permitted sequences of messages for participant detailing and comments (IS4) ... 48
Figure 32: Messages in IS4 in the context of the ECR Reference Process ... 50
Figure 33: ECR states and state transitions ... 68
Figure 34: IS1 Protocol Specification: public view of coordinator ... 74

Figure 35: IS1 Protocol Specification: public view of participant.. 75
Figure 36: IS2 Protocol Specification: public view of coordinator (without special cases).... 76
Figure 37: IS2 Protocol Specification: public view of participant (without special cases)..... 77
Figure 38: IS2 Protocol Specification: public view of coordinator (with special cases)......... 78
Figure 39: IS2 Protocol Specification: public view of participant (with special cases).......... 79
Figure 40: IS3 Protocol Specification: public view of coordinator 81
Figure 41: IS3 Protocol Specification: public view of participant.. 82
Figure 42: IS4 Protocol Specification: public view of coordinator (without special cases).... 83
Figure 43: IS4 Protocol Specification: public view of participant (without special cases)..... 84
Figure 44: IS4 Protocol Specification: public view of coordinator (with special cases)......... 85
Figure 45: IS4 Protocol Specification: public view of participant (with special cases).......... 86

Tables

Table 1: Overview of the roles of the participant in ECR Interaction Scenarios................ 38
Table 2: Recommended use of ECR Messages in the ECR Interaction Scenarios.......... 61
Table 3: Message headers and control information for ECR Messages........................... 62
Table 4: Classification for the recommended use of data objects 69
Table 5: Recommended use of data objects as content/payload for ECR Messages 69
Table 6: Cooperation-specific input parameters.. 73

1 ECR Reference Process Fundamentals

This chapter provides an overview of the fundamental principles and key phases of the ECR Reference Process. It describes the solution approaches upon which the ECR recommendation is based. The fundamental concepts and recommendations described in Part 0 of the ECM Recommendation also apply to this part of the ECM Recommendation unless specified otherwise.

1.1 ECR models and messages related to ECM

This part of the ECM Recommendation provides detailed information on the ECM Reference Sub-Process "Specification and Decision on Change" (subsequently called “ECR Reference Process”), which permits the synchronization of the different internal procedures of the involved partners using ECR Messages. It offers recommendations for reasonable sequences of ECR Messages, grouped into ECR Interaction Scenarios (IS) within the context of the overall ECM process between the cooperation partners.

Furthermore, it provides recommendations with respect to the minimum requirements applicable to the communication and ECR Message content. Messages described under "Common Resources for ECM" in Part 0 of this recommendation are referenced or specified further if required for the ECR Reference Process.

Part 0 specifies the general ECM Data Model. Within this part of the ECM Recoomendation the general ECM Data Model is applied to the ECR context, i.e. attribute values are defined in detail for the ECM Reference Process.

1.2 ECR specific rules

The global set of rules defined in this recommendation (Part 0) is also valid for the ECR Reference Process. Additional specific rules for the ECR Reference Process, ECR Interaction Scenarios and messages, as well as the ECR data have not yet been defined.

2 Communication and the ECR Reference Process

This chapter provides an introduction to the ECR Reference Process as Sub-Process of the overall ECM Reference Process. It covers the processing of an ECR from the initial proposal for a change through to its approval or rejection. The ECR Reference Process serves as a harmonized ECR process for standardized interaction between partners and forms the basis for a common understanding during communication. In Chapter 3, ECR Interaction Scenarios are derived from the ECR Reference Process which defines in detail specific roles and processing steps for the partners and the interactions for specific partnership constellations. Introductory notes on modelling are followed by a detailed description of ECR Refernence Process together with its sub-processes.

2.1 Overview of the modelling concept and the notation used

The processes and sub-processes involved in the ECR Reference Process are described below on the basis of diagrams. Separate diagrams are used for a general overview of a process and for detailed views of a process. The ECR Reference Process and its sub-processes are described from the point of view of the primary process owner (*coordinator*). Partners (*participants*) can fulfil certain roles in the process (see also section 2.3). In this case, interactions with the partner will take place. Phase diagrams are used to provide a general overview of the phases. These show a linear representation of the consecutive phases. Typically, phases are initiated when milestones are reached and, in the same way, milestones can be reached at the end of phases. Milestones represent the reaching of a positive state in the process meaning that the request to change is pursued further on. If one of the phases can be carried out by a partner, this phase is shown in a line entitled *Participant*, and the primary process is shown in the line entitled "Coordinator". If the partner does not have an active role in a phase, but only introduces or receives notifications, this is not indicated separately in the phase diagram.

Activity diagrams are used to provide a more detailed description of the processes in the ECR Reference Process. These are described in UML 2 (Unified Modeling Language 2) (see Appendix A of Part 0). In the activity diagrams, the activities model processing steps which are carried out by roles. The sequence of the activities is described in more detail. Composite activities are described in further detail in separate diagrams or are derived from patterns which are described in Appendix A of Part 0. In the case of interaction between partners, individual interactions between roles, during which message objects are exchanged, are modelled. The relationships between the roles used in the ECR Reference Process and its sub-processes are drawn in the diagram for the organization model. In particular, this shows which roles in the activity diagram are assigned to the *coordinator* and which roles can be embodied by *participants*. Specific statuses which describe the positive or negative result of an activity, for example, are modelled using specific objects.

To achieve greater clarity, a distinction is made between the normal flow of an activity and special variants of this flow and exceptions for the activity. The normal flow which is represented need not always be the usual case during actual execution; instead it describes the primary activities in a way which is easy to understand. Appendix A of Part 0 contains notes on the notation used.

The following should be noted with respect to the terms "phase" and "activity": In the description of the phase diagrams, the term "phase" is used throughout to refer to the processing steps; in the descriptions of the activity diagrams, the term "activity" is used throughout. For each of the phases described in detail, there is either one activity which has the same name, or the phase comprises a number of activities. Similarly, the milestones used in the phase diagrams have corresponding states in the activity diagrams.

2.2 Overview of the ECR Reference Process

The purpose of the phase will first be described, followed by an overview of each phase. This is then followed by more detailed descriptions of the activities in the course of the normal flow and of special flow variants and exceptions (cancellation, fast-track execution, rollback).

2.2.1 Purpose

The objective of the ECR Refernece Process is to determine the extent and impact of the planned change, its feasibility (with regard to design, manufacturing), affected persons or departments, affected parts/documents, the collection of comments and assessments by experts concerning costs, dates, quality, weight, preparation time, and others, and finally the decision to accept or reject the planned change.

2.2.2 Phases of the ECR Reference Process

In the context of the overall ECM Reference Process the *Specification and Decision on Change* phase begins after milestone M3, *Potential Solution Defined*, has been reached. It starts with the *Inquiry of ECR* phase, in which a proposal for a change is received and checked to see whether it will be pursued (see Figure 1). If the change proposal is to be pursued further, milestone M3.1, *ECR Initiated*, is reached and the proposal for a change is created as a change request in the *Creation of ECR* phase and is prepared for further evaluation.

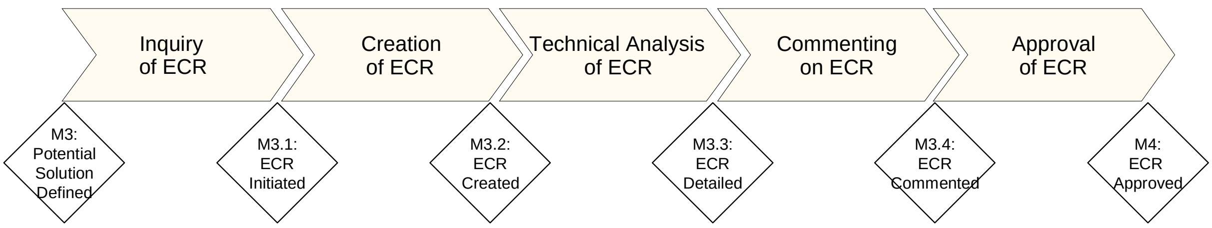

Figure 1: Phases of the ECR Reference Process

If the ECR is to be pursued further, then milestone M3.2, *ECR Created*, is reached and technical analysis and evaluation of the ECR is carried out in the *Technical Analysis of ECR* phase. After finishing the analysis milestone M3.3, *ECR Detailed*, is reached and the ECR is evaluated from the point of view of other specialist departments in the *Commenting on ECR* phase. After finishing commenting milestone M3.4 *ECR Commented*, is reached, and a decision is made with respect to the ECR in the *Approval of ECR* phase. If the ECR is approved, the *Specification and Decision on Change* phase ends and milestone M4, *ECR Approved*, of the overall ECM Reference Process is reached.

2.2.3 Normal flow and status

The ECR Reference Process starts with the status ‘pending’ when a proposal for an ECR is received from one of the possible initiators of an ECR in the activity *Inquiry of ECR* (see Figure 2).

The proposal for an ECR may be accepted or rejected (documented in ACCEPTANCE.decision as shown). In the case of rejection, the ECR Reference Process is terminated with the status ‘disapproved’ (documented in the value STATUS.status), otherwise the intermediate status ‘initiated’ is reached. This is followed by the activity *Creation of ECR*, in which the regular change request is created, described in more detail and prepared for the subsequent steps. This results in a decision on whether the ECR is accepted or rejected for further detailing, commenting and approval. If the ECR is rejected, the ECR activity is again terminated in the status ‘disapproved’; otherwise the intermediate status ‘created’ is reached. *Creation of ECR* is followed by the activity *Technical Analysis of ECR*, in which the technical aspects and the impact of the potential change are examined in greater detail and evaluated. In particular, the scope of the change is determined with respect to the parts/documents affected etc. Once the change request has been described in detail, the intermediate status ‘detailed’ is reached and the activity *Commenting on ECR* follows, in which the change request is evaluated from other points of view. Once this activity is completed, the intermediate status ‘commented’ is reached, and the ECR is prepared for the activity *Approval of ECR*, in which the change request is finally accepted for implementation and the process reaches the status ‘approved’. Alternatively, the *ECR* can ultimately be rejected, in which case the status ‘disapproved’ is again reached.

The individual activities are described in more detail in section 2.4.

From the time the activity *Technical Analysis of ECR* starts until the time the activity *Commenting on ECR* ends, the list of those involved in the ECR Reference Process can be changed (several times if necessary) by carrying out the optional activity *Modify Distribution List.* Depending on the results of the activities *Technical Analysis of ECR* or *Commenting on ECR,* participants can be added, for example to process analyses or make comments, or removed because they are not affected by the change.

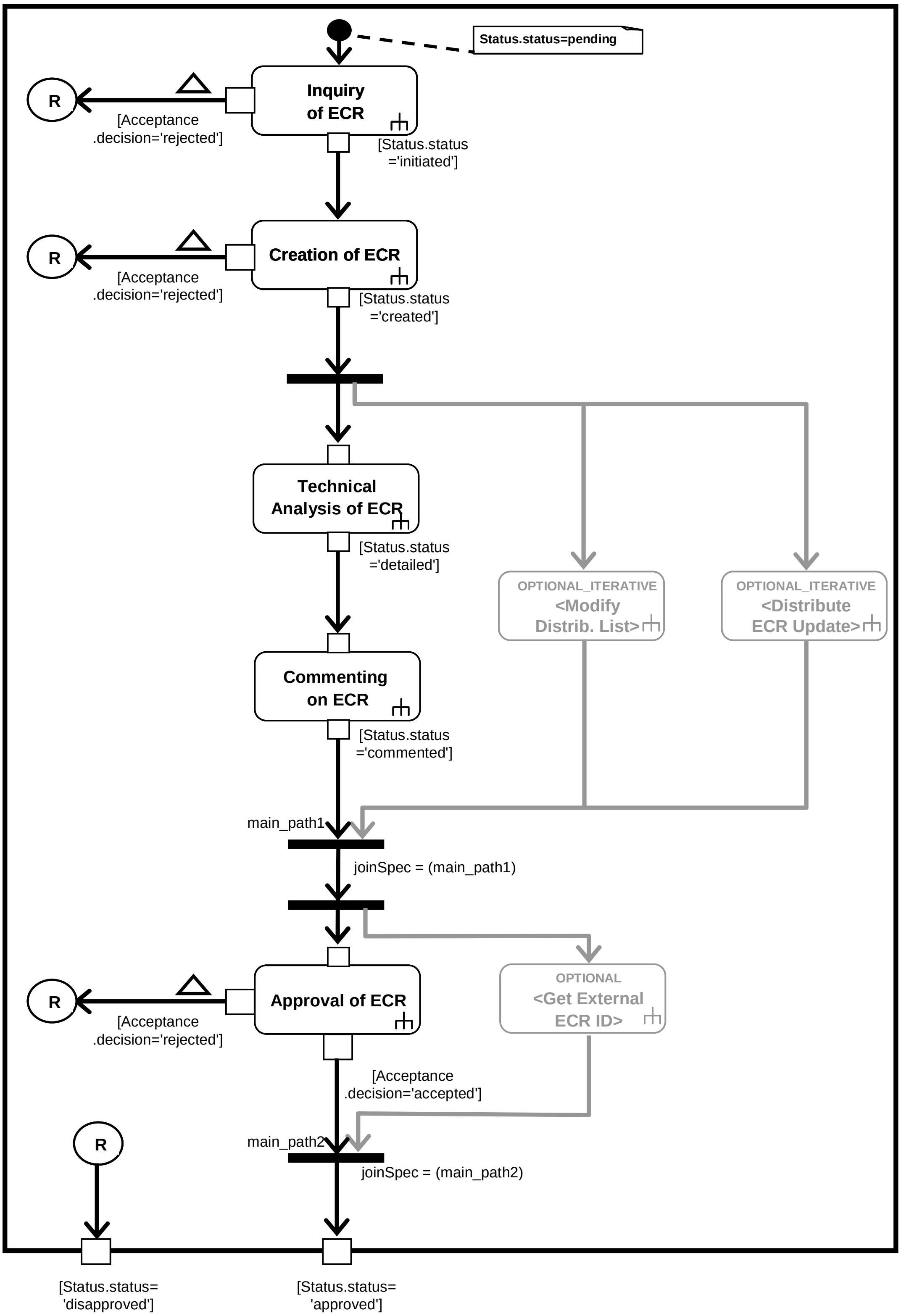

Figure 2: Normal flow of ECR Reference Process

In addition, from the time the activity *Technical Analysis of ECR* starts until the time the activity *Commenting on ECR* ends, a list of those involved in the process (notification list) can be optionally notified of any updates to the ECR by carrying out the activity *Distribute ECR Update* (see also Figure 3). To do this, a subset of those involved in the process can be selected in the activity *Select Notification List* and the message *Notify_ECR_update* containing updated ECR data can be sent to the selected people.

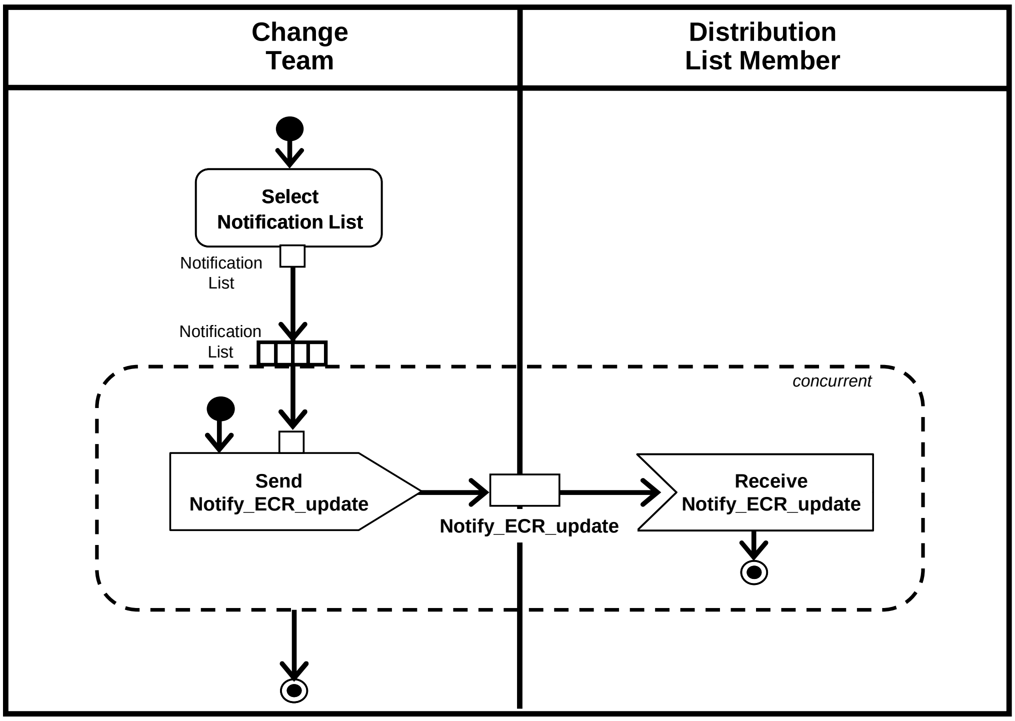

Figure 3: Activity Distribute ECR Update

If external partners are requested to approve the change request during the *Approval of ECR* phase, these partners can transfer the ECR identifier of the partner if this has been agreed using the message *Notify_ECR_id_assigned*. This identifier is then used in subsequent communication with the partner (see Figure 4).

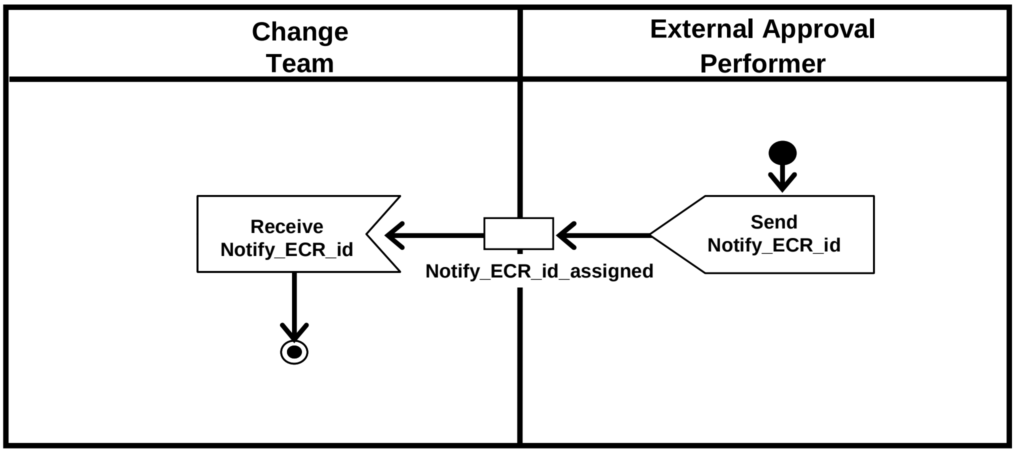

Figure 4: Activity Get External ECR ID

2.2.4 Special flow variants and exceptions

In addition to the linear process flow described above, it is also possible to exit the process early, to carry out fast-track execution and to roll back the flow. These variants are described below and are partly illustrated in Figure 5.

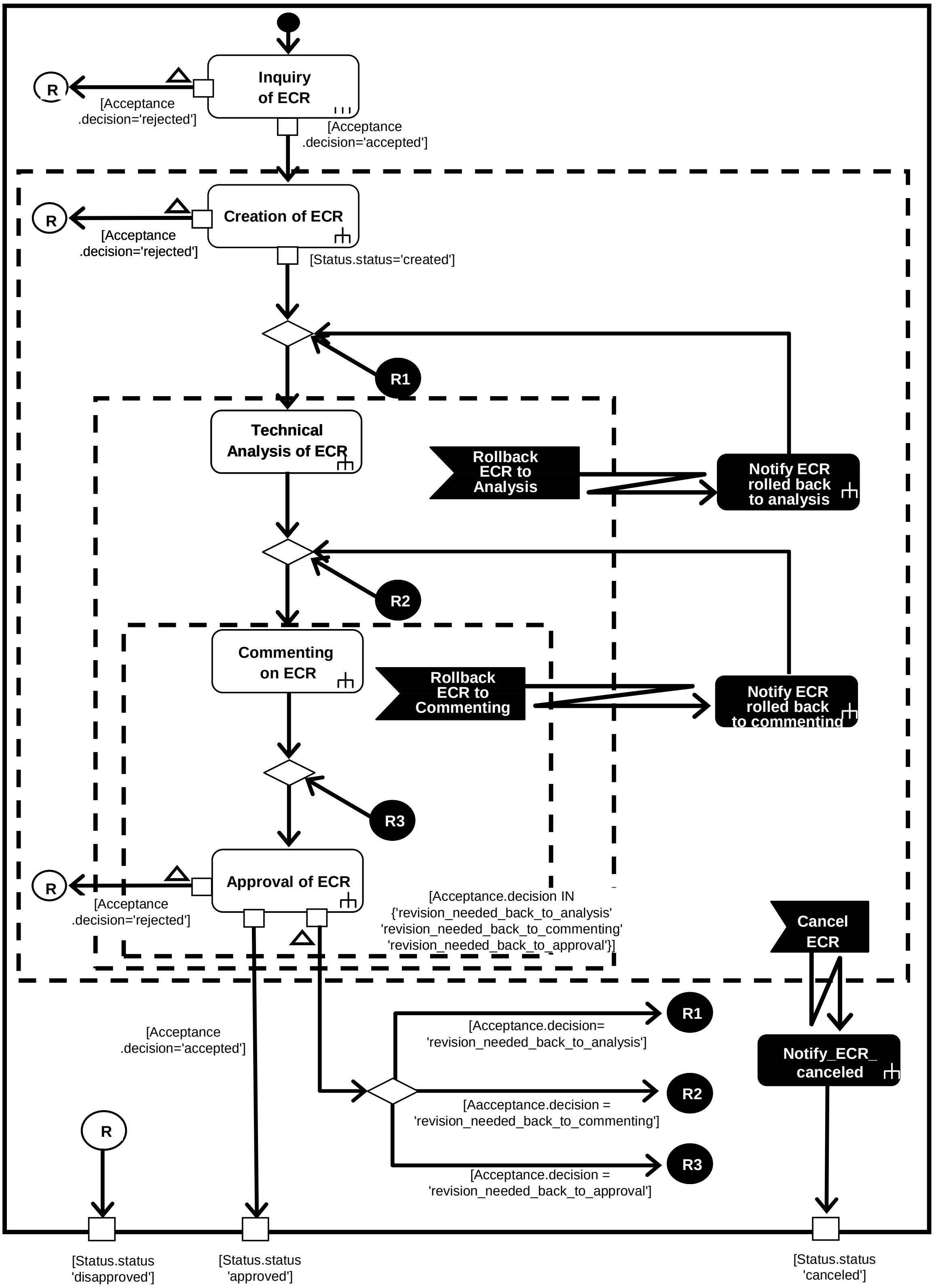

Figure 5: Flow variants of the ECR Reference Process

Cancellation of the ECR

The ECR Reference Process can be canceled by means of *Cancel ECR* during the activities *Technical Analysis of ECR* through *Approval of ECR* if the planned change no longer appears to have sufficient potential benefit during the course of the process, for example as a result of outlay or deadline considerations. In this case, persons on the *Distribution List* can be notified of this by means of the *Notify_ECR_canceled* message (see Figure 6).

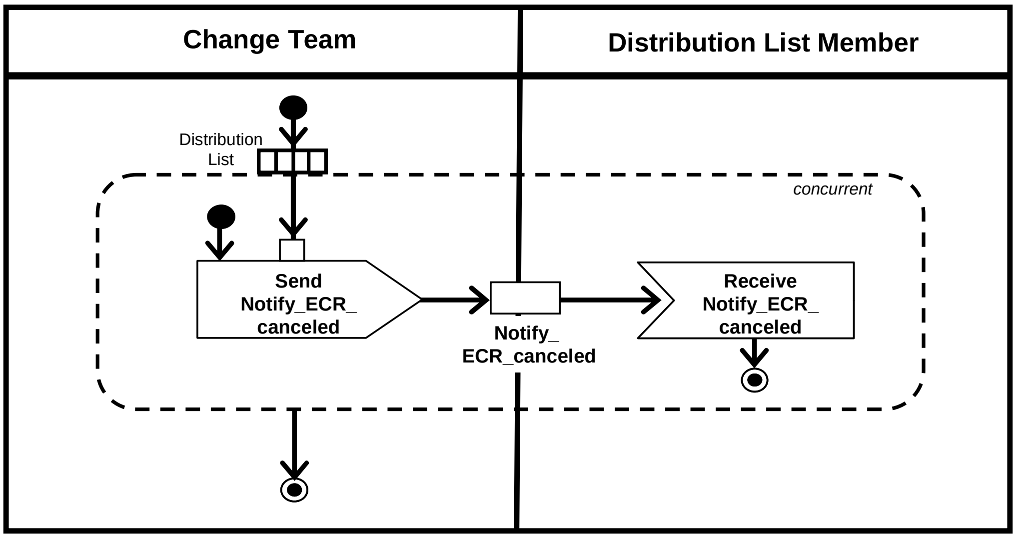

Figure 6: Activity Cancel ECR

Fast-track execution

Fast-track execution is a variant of the ECR Reference Process in which special measures are taken compared with the normal process with the objective of reducing the throughput time of the ECR or of implementing the change earlier than usual. The following types of fast-track execution are recommended:

1. Fast run: This variant does not involve a special process; those responsible for performing the activities simply execute them more quickly than in the normal process.
   This variant is indicated by a CLASS instance with CLASS.id = 'fast run' referenced by HEADER.change_process_type.

2. Estimation-based approval: With the "estimation-based approval" variant, it is laid down in the first three activities (*Inquiry of ECR*, *Creation of ECR* or *Technical Analysis of ECR*) that the decision regarding the ECR will be taken on the basis of the estimated value as specified in the activity *Technical Analysis of ECR* and that review-evaluation comments during the activity *Commenting on ECR* (see 2.4.4) will be skipped. In this case, only technical comments etc. are exchanged during *Commenting on ECR*. Estimation-based approval can be selected if, in the opinion of the *EC Manager*, the estimated evaluation of the ECR from the activity *Technical Analysis of ECR* is adequate or if time constraints necessitate a shorter throughput time for the ECR.
   This variant is indicated by a CLASS instance with CLASS.id = 'estimation based approval' referenced by HEADER.change_process_type.

3. Requests approved in advance: The early start of the activity *Realization of Change*, in particular if provisional approval of the ECR is available at the beginning of the ECR Reference Process, represents a further identified variant of fast-track execution. This variant of the *Specification and Decision of ECR* process can become necessary if the conditions associated with the change to be evaluated, for example a problem which must be solved urgently, make it inevitable that this change will be approved and that implementation must start as soon as possible. A description of this flow was outside the scope of the current version of this recommendation, but is expected to be included in a subsequent version.

4. Other variants. Other variants are "special processes" within change management, such as extraordinary authorization, which are not automatically associated with a modified process. Variants such as this are indicated by a CLASS instance with CLASS.id = 'deviation' referenced by HEADER.change_process_type.

Rollback and revision

A rollback describes a situation in which a process is interrupted and is (partially) rolled back in order to repeat certain activities. A rollback is generally carried out at any place during an activity which is in progress, and the restart point is at the beginning of a previous activity. A revision of the ECR occurs when an ECR is decided during Approval of ECR and the result is neither approval nor rejection but the need for a revision after part of the ECR Reference Process has been repeated.

***Note**: A distinction between rollbacks and revision is made due to the following difference. When a rollback is executed, there may be activities currently being executed, e.g. commenting on the ECR by the partner, which are aborted. In contrast, revision takes place at a dedicated and planned point in the ECR Reference Process where no activities need to be aborted.*

Revision as used here does not imply that there is a need for versioning of the ECR. In fact, versioning is not recommended.

The following types of rollback and revision may occur.

1. Rollback R1: The ECR can be rolled back to the start of the activity *Technical Analysis of ECR* during the activities *Technical Analysis of ECR, Commenting on ECR* through *Approval of ECR*. It may, for example, be necessary to roll back to the start of the activity *Technical Analysis of ECR* if during subsequent processing of the ECR it becomes clear that the technical analysis which is currently being carried out or which has already been carried out is/was insufficient or if new information with respect to the conditions that apply becomes known with the result that it is necessary to restart the analysis. In this case, the activity *Notify ECR rolled back to analysis* (see Figure 7) can be used to specify a selection of the affected participants in the process and inform them that the ECR Reference Process has been rolled back with the message *Notify_ECR_rolled_back_to_analysis*.

2. Rollback R2: The ECR can be rolled back to the start of the activity *Commenting on ECR* during the activities *Commenting on ECR* through *Approval of ECR*. A rollback of this type may be necessary if during the subsequent course of *Specification and Decision of ECR* it

becomes clear that the commenting of the ECR which is currently being carried out or which has already been carried out is/was insufficient or if new information with respect to the conditions that apply becomes known with the result that it is necessary to restart the commenting procedure. In this case, the activity *Notify ECR rolled back to commenting* (see Figure 8) can be used to specify a selection of the affected participants in the process and inform them that the process has been rolled back with the message *Notify_ECR_rolled_back_to_commenting*.

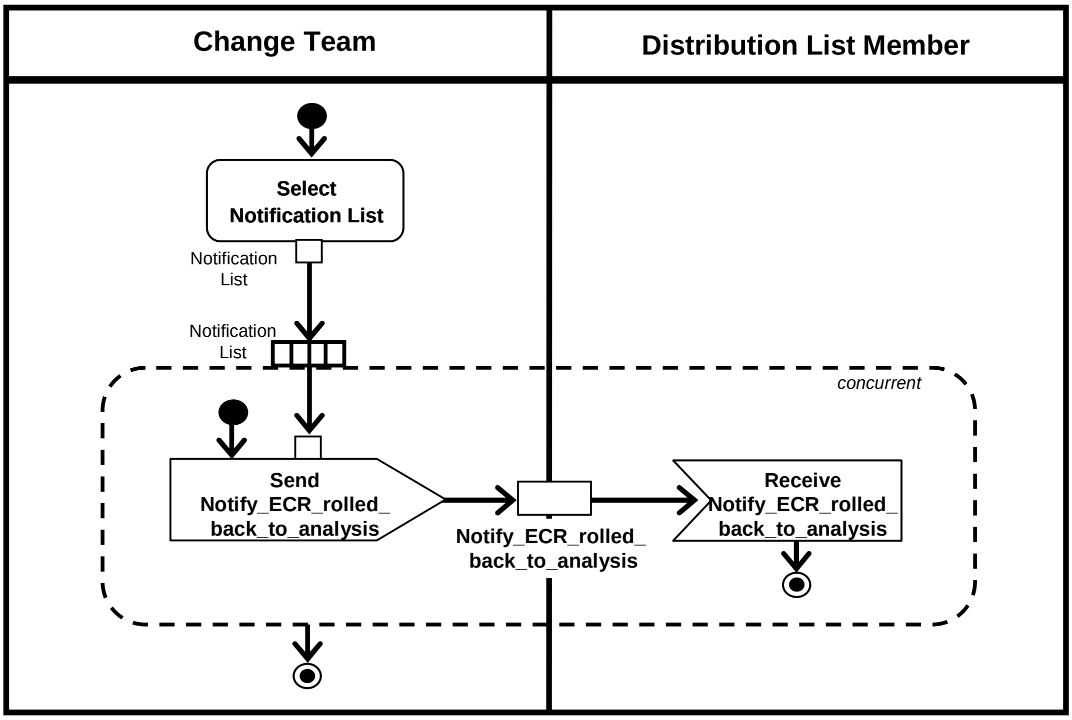

Figure 7: Activity Rollback ECR to Analysis

3. Revision: The activity *Approval of ECR* can result in the need to revise the ECR if the activity has resulted in one of the following values (documented in ACCEPTANCE.decision).
   a) 'revision_needed_back_to_analysis': In this case, the process is rolled back to the beginning of the activity *Technical Analysis of ECR*. This result can occur if, while making a decision on the ECR, it is established that certain aspects must be additionally included or that the scope must be enlarged, or that these issues must be re-examined during the *Technical Analysis of ECR*.
   b) 'revision_needed_back_to_commenting': In this case, the process is rolled back to the beginning of the activity *Commenting on ECR*. This result can occur if, while making a decision on the ECR, it is established that certain comments must be additionally included or re-drafted during evaluation.
   c) 'revision_needed_back_to_approval': In this case the process is rolled back to the beginning of the activity *Approval of ECR*. This result can occur if, while making a decision on the ECR, it is established that the ECR must be resubmitted, for example for formal reasons or if an external approval (see 2.4.5) needs to be performed

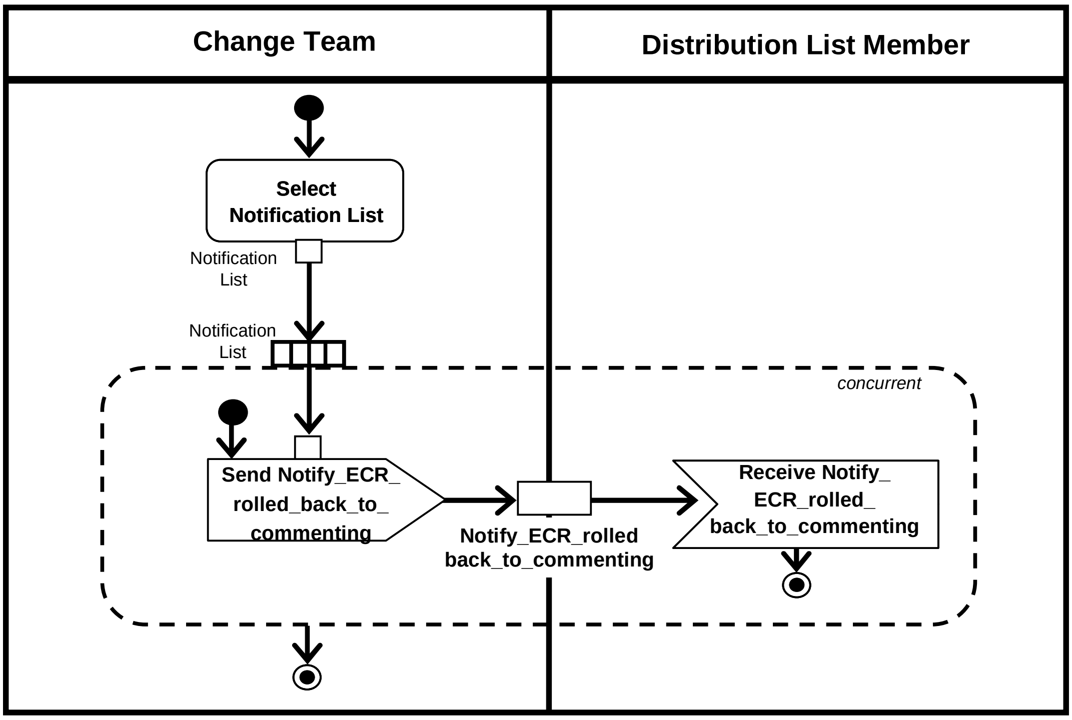

Figure 8: Activity Rollback ECR to Commenting

2.3 ECR organization model and roles

Certain organizational elements form the basis for the ECR Reference Process. These describe contiguous scopes of responsibility within the ECR Reference Process and are independent of concrete persons or organizational units. Within the ECR Reference Process they appear as the performers of activities or as senders or recipients of notifications. In addition, some of them occur in the data model in order to communicate concrete persons or concrete organizational units (see 4.2). In this context, an organizational unit indicates an element of an organizational structure such as a company, department, etc. Organizational entity is the collective term for concrete persons and concrete organizational units. Figure 9 below shows a diagram in which the various organizational elements occur and which indicates the relationships between them.

A distinction is made between the following three types of organizational elements, each of which is indicated by a special symbol in the organization diagram.

- **Organizational unit role:** This is a placeholder for an organizational unit. This is the point at which the *coordinator* and the *participant* occur in the organization diagram. *Coordinator* is a company which has responsibility for an ECR. *Participant* is a company which is involved in an ECR process belonging to a *coordinator*.
- **Role:** Placeholder for an organizational entity (person/organizational unit) which performs certain tasks and activities in one or more processes, which has a defined scope of responsibility and which possesses the appropriate competence. The roles are shown at the bottom of the diagram. Internal roles are only played by persons/institutions of the *coordinator* and potentially external roles can also be played by the *participant* or persons/institutions of the *participant*.

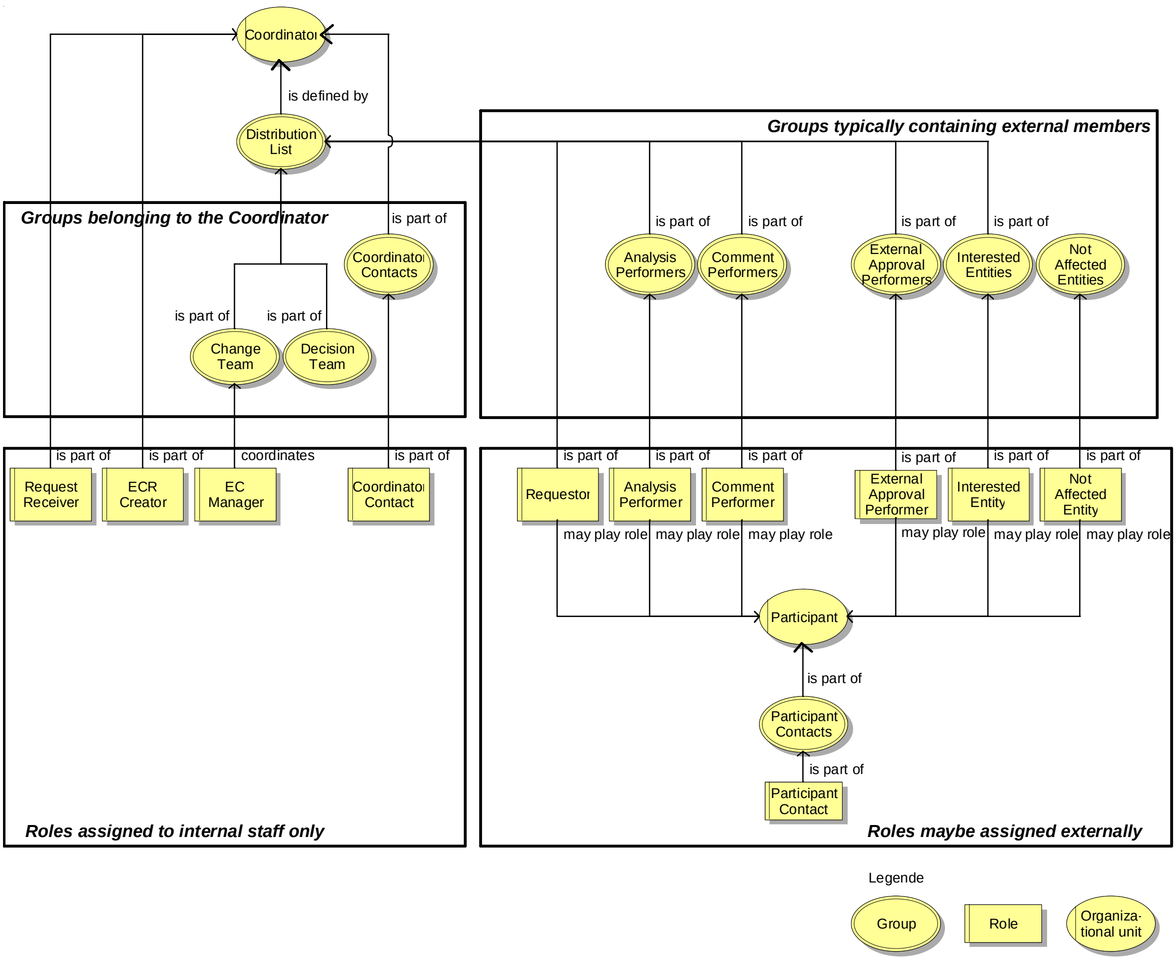

Figure 9: ECR organization model

- **Group:** Set of organizational entities (persons/organizational units) with a particular role. The groups are shown at the top of the diagram. A distinction is made between two types of groups. 'Groups belonging to the *Coordinator'* are groups in the ECR Reference Process which have roles mostly played by organizational entities within the *coordinator* (Please note: These groups may also contain external members, but the group in general resides at - and therefore communicates from the side of - the *coordinator*.) 'Groups typically containing external members': Groups in the ECR Reference Process which have roles that can also be played by a *participant*. These groups can also be empty if there is no external partner involved, or the groups can be made up of internal organizational entities. The Distribution List is a special group. This is a collective group and contains members of all groups involved in the ECR together with information on the role the organizational entities play. Members of this group are called Distribution List Members. The Distribution List is used in the ECR Reference Process when messages about the ECR Reference Process are to be issued which are of relevance to several or all groups. The Distribution List is also used as a collective term for all groups and the roles assigned to their members.

The following describes the roles and groups of the organization model in more detail.

Roles assigned to internal staff only:

- The *Request Receiver* receives proposals for changes, prepares them and decides whether or not to create a corresponding ECR.
- The *ECR Creator* creates the ECR when it has been decided that such a ECR will be created on the basis of a proposal for a change.
- The *EC Manager* (Engineering Change Manager) is responsible for coordinating the ECR, i.e. for carrying out the analysis and evaluation and for making a decision on the ECR. He is a member of the *Change Team* that he coordinates.
- A *Coordinator Contact* is a person on the *coordinator* side who can serve as a contact for one or more *participants* with respect to the ECR or a special range of issues or subjects relating to the ECR (such as set of modules/parts, design, prototyping, certification, procurement, etc.). This can be used during data exchange in order to inform the partner who is available as a contact person.

Roles that may be assigned externally:

- A *Requestor* is a person or organizational unit who introduces a proposal for a change.
- An *Analysis Performer* is a person or organizational unit who carries out a technical analysis of the ECR. Internally, this role is typically played by developers.
- A *Comment Performer* is a person or organizational unit who carries out an evaluation of the change and then gives their comment of the ECR. Internally, this role is typically played by experts for a specialist evaluation of the ECR, for example experts from procurement, logistics, prototyping etc.
- An *External Approval Performer* is an external organizational unit requested to approve the change from their point of view during the *Approval of ECR* phase.
- An *Interested Entity* is an external organizational unit interested in the progress or result of an ECR and is therefore informed of any important changes to the status of the ECR.
- A *Not Affected Entity* is a person or organizational unit initially involved because they were potentially affected, but during the course of the ECR are no longer regarded as being affected. This role allows such a situation to be documented in order to allow the person or organizational unit to still be mentioned during communication.
- A *Participant Contact* is a person or organizational unit who serves as a contact with respect to the ECR or a special range of issues or subjects relating to the ECR (such as set of modules/parts, design, prototyping, certification, procurement, etc.). This can be used during data exchange in order to inform the *coordinator* who is available as a contact person.

Groups belonging to the *Coordinator*:

- The *Change Team* group is a group of persons or organizational units which coordinate and pursue a specific ECR process. Among other things, the *Change Team* defines who is to be involved in the ECR process for analysis, evaluation or information purposes and, for certain ECRs, is able to make preliminary decisions. The *Change Team* can comprise only the *EC Manager* or, for example, one or more interdisciplinary teams responsible for the ECR.

- The *Decision Team* group is a group of persons or organizational units who make a decision on the ECR during the *Approval of ECR* phase. This team typically comprises senior managers.
- The *Coordinator Contacts* group contains all the persons who have the role *Coordinator Contact*.

Groups typically containing external members:

- The *Analysis Performers* group contains all the persons/organizational units who have the role *Analysis Performer*.
- The *Comment Performers* group contains all the persons/organizational units who have the role *Comment Performer*.
- The *Interested Entities* group contains all the persons/organizational units who have the role *Interested Entity.*
- The *Not Interested Entities* group contains all the persons/organizational units who have the role *Not Interested Entity*.
- The group *External Approval Performers* contains all the persons/organizational units who have the role *External Approval Performer*. If this group contains any members, they are all external.
  - The *Participant Contacts* group contains all the persons/institutions that have the role *Participant Contact*. If this group contains any members, they are all external.

Note: *Which of the roles mentioned need to be communicated to the partner depends on the cooperation involved.*

2.4 ECR Reference Process

In the following sections, the five phases of the ECR Reference Process are described in detail: *Inquiry of ECR, Creation of ECR; Technical Analysis of ECR, Commenting on ECR and Approval of ECR.* The purpose of the phase will be described, followed by an overview of each phase. This is then followed by more detailed descriptions of the normal process and of special flow variants and exceptions.

Note**:** *The optional ECR Message Notify_response_deferred is not explicitly illustrated in the following process diagrams of the ECR Reference Process. This ECR message of the recipient of a previously received Request Message (e.g. Request_ECR_comments) informs the sender of that Request Message that the Respond Message to the original Request Message will be deferred*

2.4.1 Inquiry of ECR phase

This section provides a detailed description of the *Inquiry of ECR* phase.

Purpose

The *Inquiry of ECR* phase serves to receive proposals for ECRs. These proposals may originate from development, production, sales and other specialist departments or from partners (suppliers, development partners, etc.). Proposals for changes are aimed at reducing costs, improving quality, adapting to changes in the production process, responding to statutory constraints (environment, safety) or at implementing customer wishes with respect to new functionality, special equipment, etc.

In this phase, at least a rough outline of the proposal for an ECR is recorded and classified, and on this basis, a decision is taken as to whether the proposal is to be pursued further as an ECR in subsequent phases. This is designed to exclude proposals for ECRs which have in the past already proved to offer no potential benefit, for example. On the basis of the rough description and the assignment of an ECR, a member of staff is named who will be responsible for creating one or more ECR for the named proposal and describing it in more detail in the subsequent *Creation of ECR* phase.

Overview of phase

The *Inquiry of ECR* phase starts when milestone M3, *Potential Solution Defined*, is reached. This is the point at which the change proposal, optionally including a potential solution, has been defined in the framework process, and it begins with the *Inquire Initial ECR* phase in which the proposal for an ECR, a so-called initial ECR, is received from within the company or from a partner. The proposal for a change is created for the purposes of further examination and documentation in the *Create Initial ECR* phase. Following this, those who may potentially be affected in the company or in partner companies can be notified of the fact that the initial ECR has been created in the *Notify about Creation* phase. This is followed by the *Analyze & Decide Initial ECR* phase in which the initial ECR is examined and a preliminary decision is taken as to whether the proposal will be pursued. If the initial ECR is to be pursued, milestone M3.1, *ECR initiated,* is reached and the phase is completed.

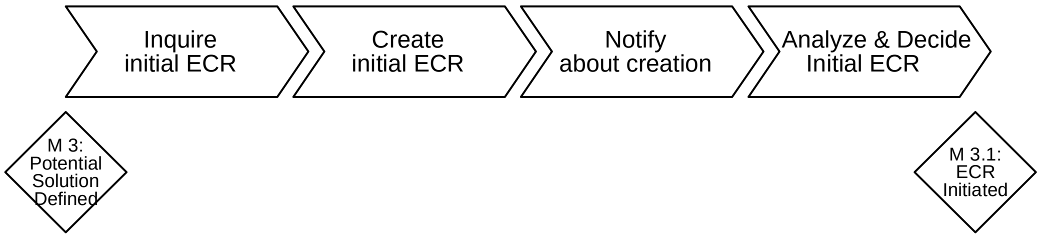

Figure 10: Phase overview for Inquiry of ECR

Normal flow

The *Inquiry of ECR* activity starts with a *Requestor* sending the message *Request_initial_-ECR* to the *coordinator* (see Figure 11). A *Request Receiver* at the *coordinator receives* the message *Request_initial_ECR* and creates one or more initial ECR in his ECM system in the *Create Initial ECR* activity for the purpose of subsequent analysis, evaluation and documentation.

He then sends a response message *Respond_initial_ECR* to the *Requestor*, to confirm receipt of the initial ECR and to notify the partner of the reference to the initial ECR to allow the partner to address queries and monitor the initial ECR. The *Request Receiver* then prepares the initial ECR in the *Prepare Initial ECR* activity by adding any necessary information and a description, such as classification criteria, description of the cause etc. The *Request Re-*

ceiver then examines the proposal for a change in the *Analyze Initial ECR* activity, for example taking account of similar ECRs from the past or current change requests with a similar scope. The *Request Receiver* can then decide whether the initial ECRis to be pursued as an ECR or not in the subsequent activity *Decide Initial ECR*. If the proposal is not to be pursued (i. e. ACCEPTANCE.decision = 'rejected') the *Request Receiver* then sends the *Requestor* the message *Notify_initial_ECR_rejected*, in order to inform him of the decision that the proposal is not to be pursued and to give reasons for this decision. Otherwise (i.e. ACCEPTANCE.decision = 'accepted') the activity is finished. In this case, notification takes place in the following phase *Creation_of_ECR*.

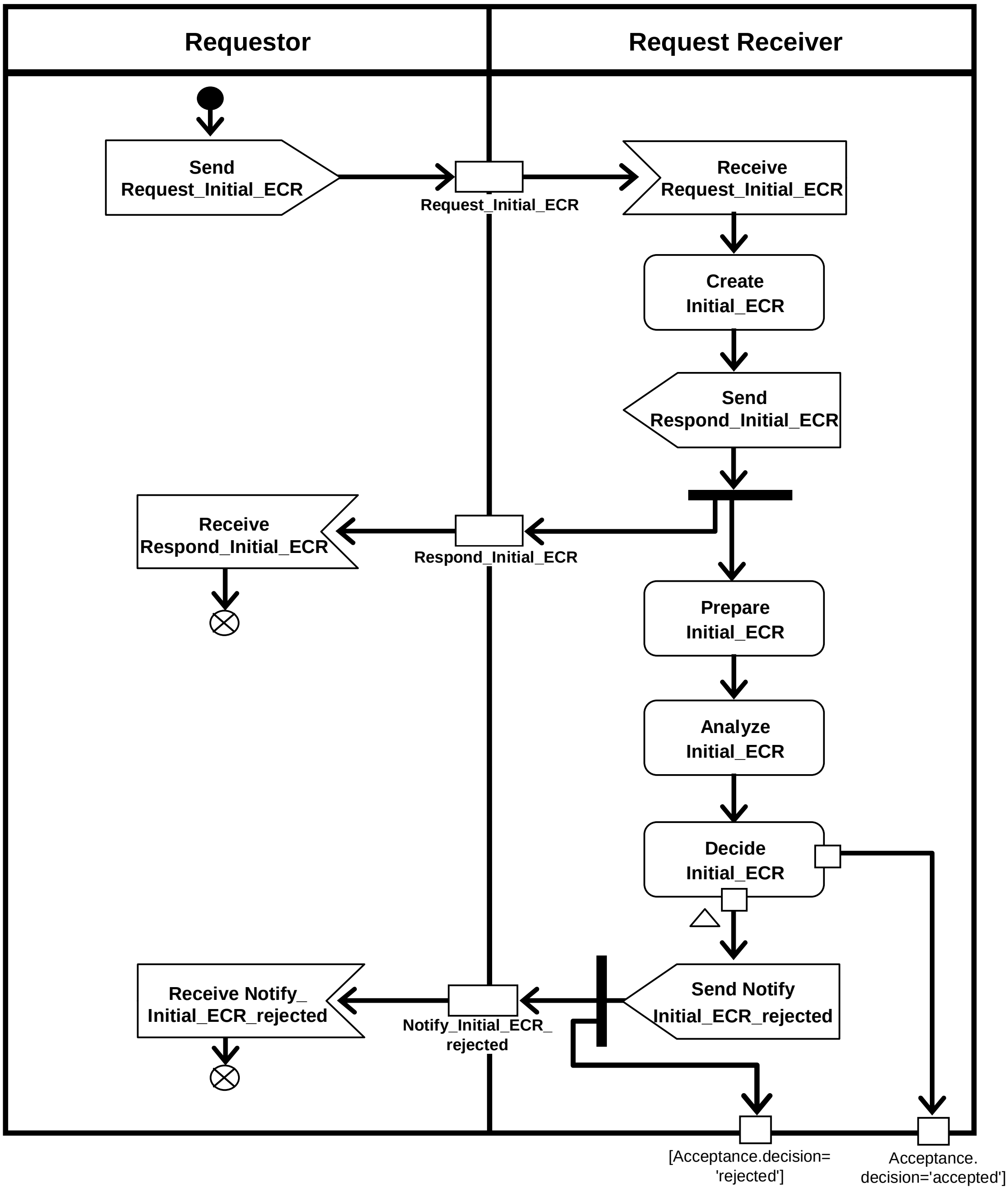

Figure 11: Normal flow for Inquiry of ECR

Special flow variants and exceptions

No special flow variants and exceptions are described.

2.4.2 Creation of ECR phase

This section provides a detailed description of the *Creation of ECR* phase.

Purpose

After the proposal for an ECR has been received and a decision has been made to create an initial ECR in the preceding phase *Inquiry of ECR*, the ECR is actually created and described in more detail in this phase. The ECR is assigned a unique ID, is classified and is provided with a general description. This involves ECR data such as the problem description, the scope of the change (series, projects, etc. that are affected), the proposed solution for the change, the Distribution List etc. being supplemented and described in more detail. This is frequently done by a developer. The description can be carried out in consultation with the *Requestor* of the ECR in order to receive more precise information on the initial ECR. After this more detailed description has been completed, the ECR can be rejected if it appears to offer insufficient potential benefit or if current priorities mean that it needs to be deferred. If the ECR is pursued, it is prepared in such a way by the end of the phase that the general framework of the change has been outlined for the subsequent superordinate *Technical Analysis of ECR* phase. In addition, those responsible for carrying out the subsequent phases, *Technical Analysis of ECR, Commenting on ECR and Approval of ECR*, can be defined in this phase.

Overview of phase

The *Creation of ECR* phase starts when milestone M3.1, *ECR initiated*, has been reached (see Figure 12).

Figure 12: Phase overview for Creation of ECR

In order to implement this decision, an ECR is created on the basis of the initial ECR from the first phase. This is described in more detail in the subsequent phase *Describe ECR & Define Change Team* in which the members of the *Change Team* including the *EC Manager* who are responsible for pursuing the ECR are defined. The *Change Team* then specifies the ECR in detail in the *Specify ECR & Distribution List* phase. At the same time, the decision is taken as to who will be actively involved in the ECR during the course of the ECR and who will be informed of developments both in the company itself and in partner organizations. In the following phase, *Decide on ECR & Document Decision*, the *Change Team* takes a preliminary decision as to whether the ECR is to pass through the subsequent phases *Technical Analysis of ECR* or *Commenting on ECR*. In all cases, the requestor of the proposed change is notified about the decision, and, if the ECR is accepted, those in the company or in partner organizations who are involved in the ECR process and included in the Distribution List are notified of the creation in the subsequent phase *Notify Distribution List*. If the decision about the ECR was positive, milestone M3.2, *ECR Created*, is reached at the end of this phase.

Normal flow

The *Creation of ECR* phase starts with the activity *Create ECR*, in which a member of staff on the *coordinator* side plays the role of *ECR Creator* and creates a new ECR in his own ECM system on the basis of the existing initial ECR. This documents the change under consideration and allows subsequent examination (Figure 13).

After this, the *Requestor* is notified that the *Initial ECR* has been created as an regular ECR and is informed about the ID assigned to this regular ECR. This is done by sending a *Notify_initial_ECR_accepted* message. Following this, the *ECR Creator* describes the ECR in greater detail for subsequent processing in the activity *Describe ECR*. The *ECR Creator* then defines the *Change Team* in the activity *Define Change Team*. This *Change Team* will be responsible for coordinating and performing the ECR process in the subsequent activities related to the ECR. In particular, this involves specifying the *EC Manager* as the member of the *Change Team* who has responsibility. In the first activity, *Specify ECR & Distribution List*, the *Change Team* creates a more detailed description of the ECR and defines the Distribution List containing those actively involved in the subsequent activities *Technical Analysis of ECR*, *Commenting on ECR* and *Approval of ECR* and those who should at least be notified of the progress of the *Specification and Decision of ECR* process. After this, the *Change Team* takes a preliminary decision as to whether the ECR is to be pursued or whether, in the estimation of the *Change Team*, it should be rejected. This is done in the activity *Decide on ECR*. If the resulting decision documented in ACCEPTANCE.decision is ‘rejected’, the *Change Team* notifies the *Requestor* about this fact by means of the *Notify_ECR_rejected* message. Otherwise, if the decision is ‘accepted’, the Requestor is notified by means of the *Notify_ECR_accepted* message, and the other members of the Distribution List are then informed of creation of the ECR by means of the *Notify_ECR_Creation* message. In this case, the activity *Creation of ECR* is left with the status ‘created’ in STATUS.status.

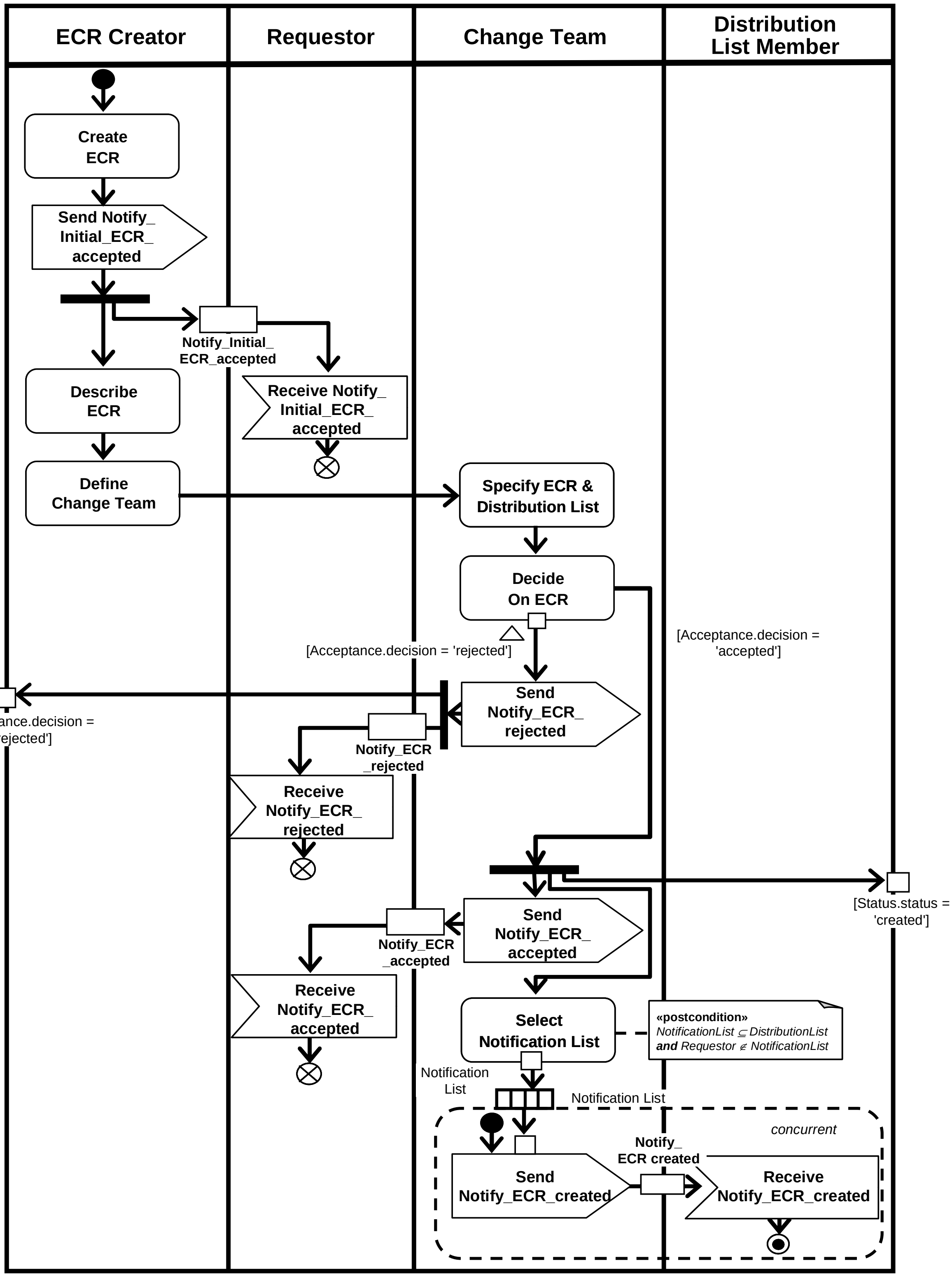

Figure 13: Normal flow for Creation of ECR

Special flow variants and exceptions

No special flow variants and exceptions are described.

2.4.3 Technical Analysis of ECR phase

This section provides a detailed description of the *Technical Analysis of ECR* phase.

Purpose

The objective of this phase is to determine technical aspects of the ECR and to gain an initial overview of its impact. Typically, the analysis is carried out by developers, each of whom perform an analysis step for the part of the ECR for which he is responsible. In doing this, the developers identify the parts or documents which are affected and give an expert opinion with respect to feasibility, implementation and any consequences of the change. In addition, they can provide estimates with respect to costs, schedule, any impact on weight, etc. as a result of the changes to the affected modules, parts or documents. Estimates with respect to the cost of the actual ECR (i.e. the costs of evaluating the change) and a schedule or individual deadlines for the ECR can also be made. Since the quantity of parts affected can increase during this phase, it is typical that the Distribution List also grows during the course of this phase. Any additional analysis staff is then also requested to draw up an analysis. This phase can also be re-entered after a rollback of the type R1 or a revision (see 2.2.4). The result of this phase is a prerequisite for any further evaluation in the subsequent phase.

Overview of phase

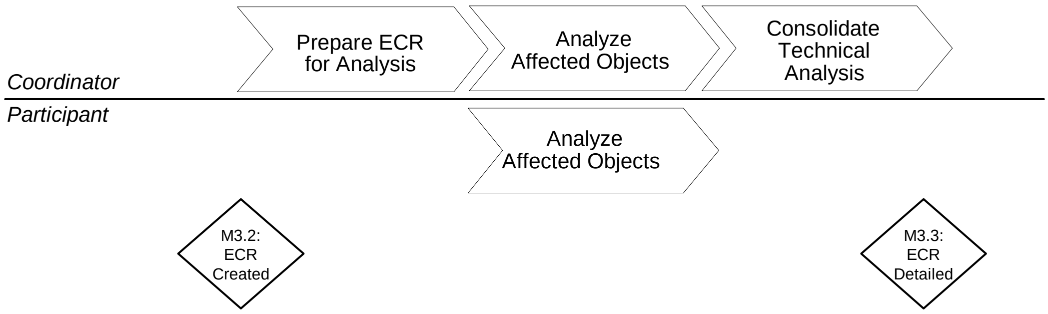

Figure 14: Phase overview for Technical Analysis of ECR

The *Technical Analysis of ECR* starts when milestone M3.2, *ECR Created,* is reached. It begins with the *Prepare ECR for Analysis* phase (see Figure 15).

After this, the *coordinator* and the *participants* involved in this phase identify the objects affected by the change (parts and documents) and examine technical aspects in the *Analyze Affected Objects* phase. After this, the results are consolidated and prepared in the *Consolidate Technical Analysis* phase. The phase ends when milestone M3.3, *ECR Detailed*, is reached.

Normal flow

In the activity *Prepare ECR for Analysis*, the *Change Team* prepares the ECR for subsequent analysis in this activity. This is done by supplementing the data required for the request to the *Analysis Performers*, such as the description of the change, the currently identified scope of parts affected by the change, etc. (see Figure 15). In particular, data specific to the *Analysis Performers* can be defined for the requests which are then sent.

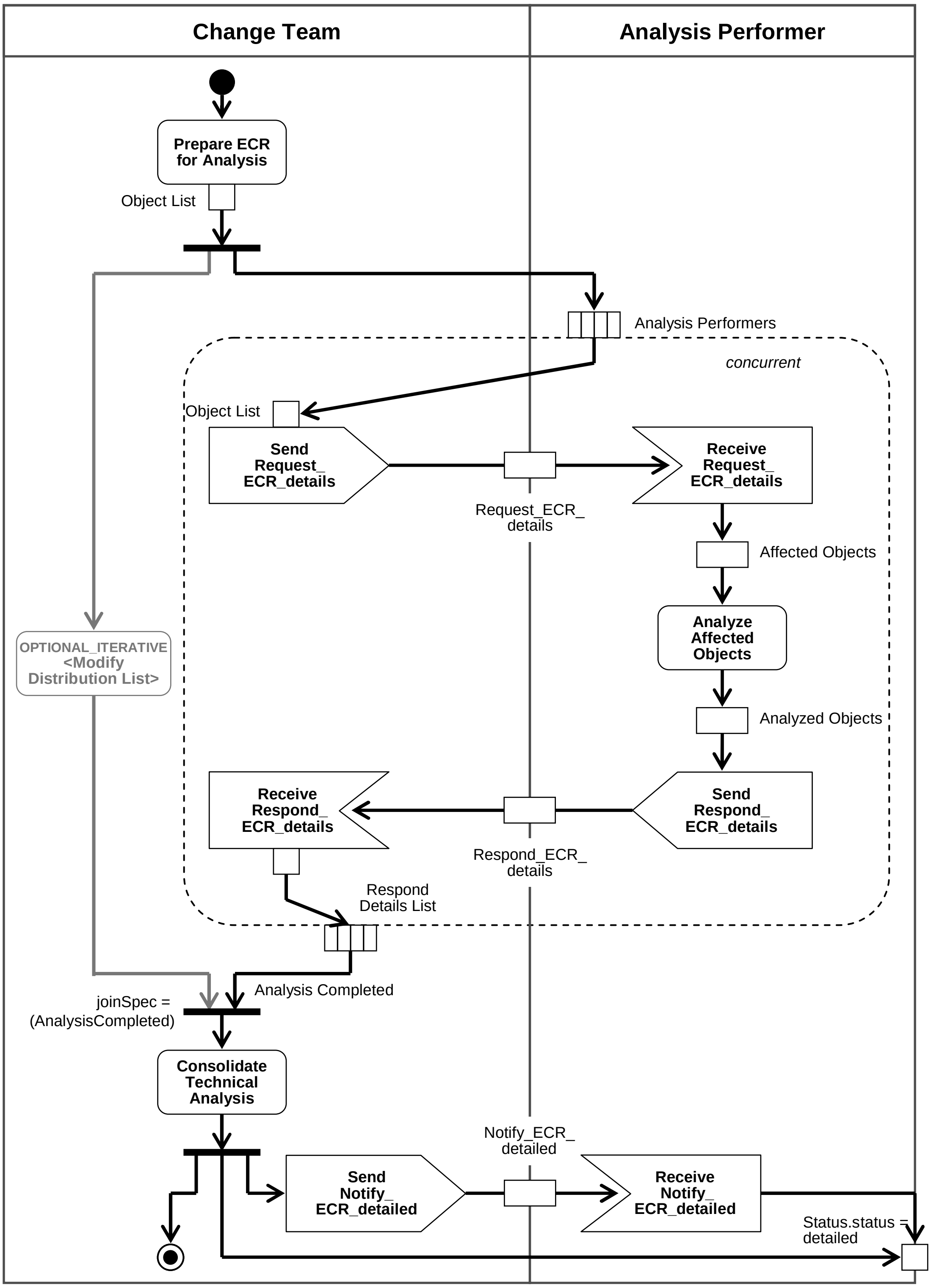

Figure 15: Normal flow for Technical Analysis of ECR

This can, for example, include the currently identified scope of parts that affect the *Analysis Performer*, specific data concerning the problem description which relates to the point of view of the Analysis Performer, etc. It can also include data requested of the partner (for example an initial estimate with respect to specific topics such as logistics etc.). The *Change Team* then sends the *Request_ECR_details* messages to the Analysis Performers for analysis, who process them in parallel.

Each message requesting analysis is received by one Analysis Performer. Each Analysis Performer analyzes the affected objects as laid down in the request he has received. The result of his analysis comprises a set of affected objects (parts, documents) describing the scope of the change and a detailed analysis of those affected objects for which the Analysis Performer is responsible. The detailed analyses contain technical statements, such as feasibility, a description of the change, estimates with respect to costs, any changes in weight, schedules etc. The Analysis Performer returns the description of the scope of the change and the detailed analyses in the response message *Respond Details.*

Note: *Here in particular it is possible that the Analysis Performer will identify and send data which should in fact only be sent in the next superordinate activity* Commenting on ECR*. In this case, this information must still be requested and returned again in the activity* Commenting on ECR*.*

Additional, affected objects lead to the need for additional analyses, which in turn means that additional Analysis Performers may need to be included. New requests are then sent to these Analysis Performers, where necessary in parallel with the other analysis requests which are in progress.

The *Change Team* collates the response messages *Respond_ECR_details* for the purposes of analysis. When the response messages have been received from all the Analysis Performers, the *Change Team* can collate and consolidate the analysis results in the activity *Consolidate Technical Analysis* and can, in particular, prepare the results for the following activity *Commenting on ECR* and the subsequent activity *Approval of ECR*. This involves supplementing or adapting master data and data from the technical analysis. In particular, a summary description of the analysis can be drafted. This completes documentation of the results of the analysis and the activity ends in the Status 'detailed' documented in STATUS.status. Optionally, the The *Change Team* may send the message *Notify_ECR_-detailed*, indicating the completion of the phase *Technical Analysis of ECR*.

Special flow variants and exceptions

Figure 16 below illustrates the possible flow variant. In order to provide information on interim results of the analysis (in the same way as is practiced in internally used ECM systems, which allow documents to be viewed by others before they are released), the *Change Team* and the Analysis Performers can use a special mechanism whereby they can repeatedly exchange *Request_ECR_details* messages and *Respond_ECR_details* messages which are flagged as incomplete. To do this, the *Change Team* can send an Analysis Performer a *Request_ECR_details* message for analysis which has been flagged as incomplete (by setting the data field IsComplete to the value FALSE (see 3.3.4)). The Analysis Performer can then use this to start work on the analysis. In this case, once the interim results are available, the *participant* can send these to the *Change Team* at the *coordinator* for analysis as a *Respond_ECR_details* message which is also flagged as incomplete.

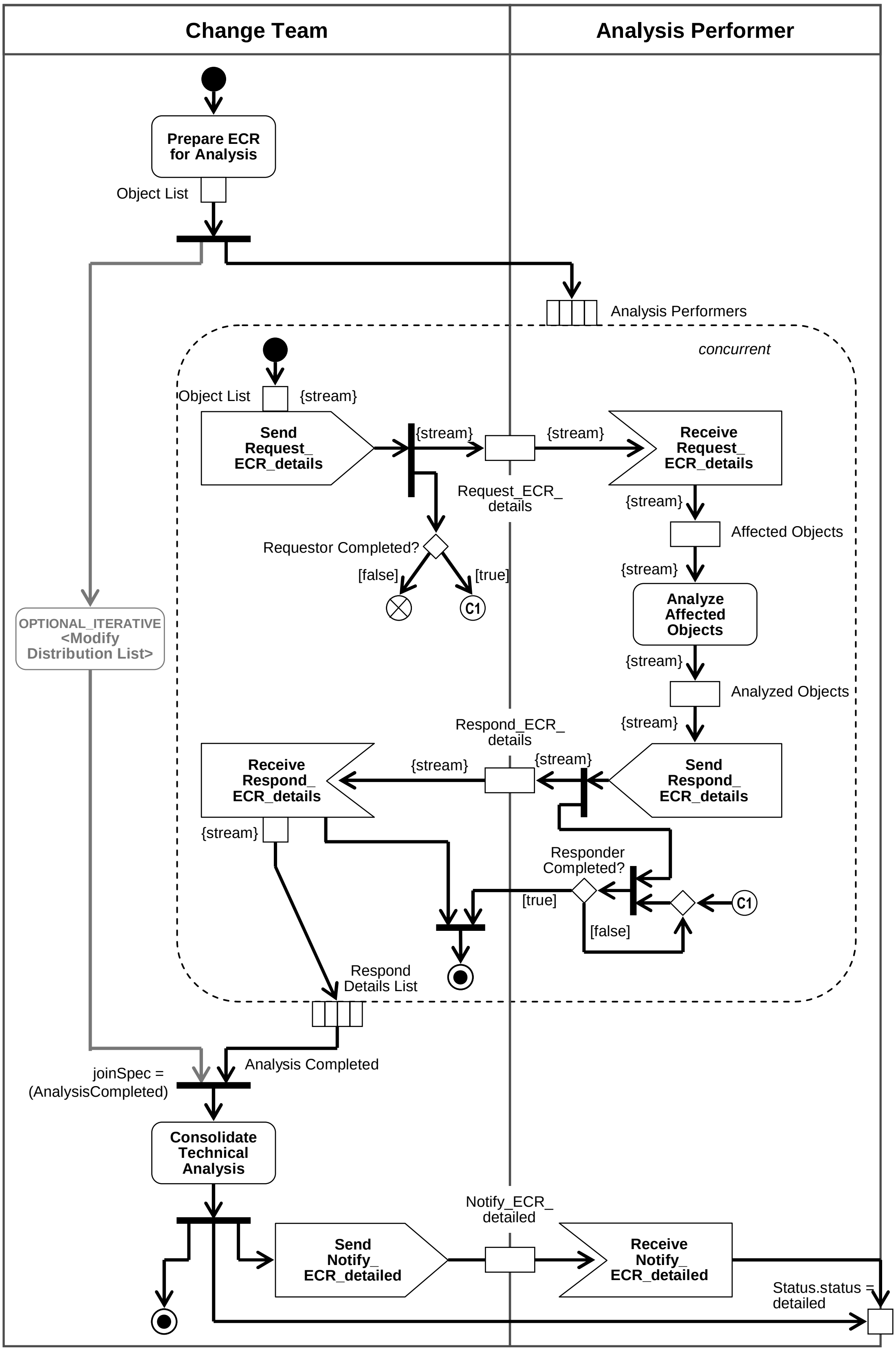

Figure 16: Flow variants for Technical Analysis of ECR

This message informs the *Change Team* of the preliminary status with respect to the scope of the change or the detailed analyses. This sequence of sending a request message which is flagged as incomplete and receiving a corresponding response message can be repeated. In this case, the reference to the corresponding request message is sent to the *coordinator* in every response message (see 3.3.4).

When the request is complete in the opinion of the *Change Team*, i.e. in particular when it contains all the affected objects for which an analysis is expected, the request is flagged as complete (by setting the data field IsComplete to the value TRUE). The *Analysis Performer* can respond to a request which has been flagged in this way using responses to the analysis which have been flagged as incomplete in order to inform the Change Team of an interim result. Finally, the *Analysis Performer* also sends a response message which is flagged as complete. This contains his completely documented analysis results.

If this procedure is not to be used, the *Change Team* only sends one request, which is flagged as complete. The *Analysis Performer* responds to this message with a single response, also flagged as complete.

2.4.4 Commenting on ECR phase

This section provides a detailed description of the *Commenting on ECR* phase.

Purpose

Whereas the *Technical Analysis of ECR* phase focused on investigating the technical feasibility of the ECR and determining the scope of the change, this phase also examines the ECR from other points of view with regard to its impact on topics such as prototyping, certification, procurement, logistics, quality assurance, etc. This is necessary in order to gain a complete picture of the impact of the ECR and its feasibility prior to the *Approval of ECR* phase. Depending on the way in which tasks are distributed in a collaborative vehicle project, partners can be requested to provide a comment on one or more specialist topics.

Overview of phase

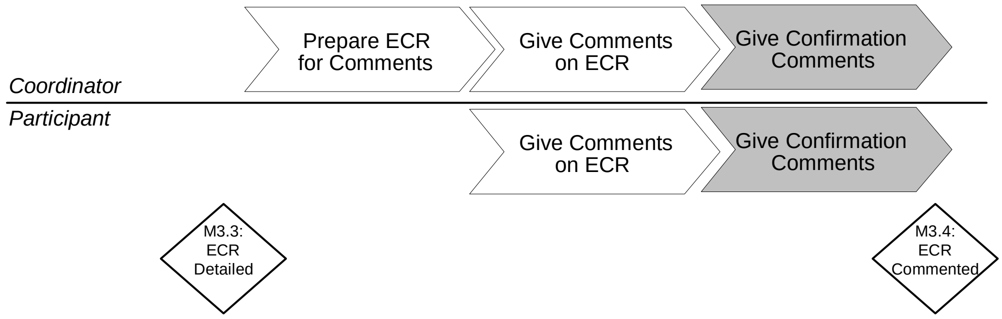

Figure 17: Phase overview for Commenting on ECR

In the previous phases, estimates are made of the extent to which optional topics may be impacted by the change and thus need to be evaluated in the form of an comment in the current phase. After the comments have been submitted by the specialist departments already mentioned, there is the option of requesting or submitting a confirmation of the validity of the comment in the light of the insights gained from the comments from the other specialist areas. The final result of this phase is an ECR which has been evaluated from the point of view of all the relevant specialist departments and which has been prepared for the *Approval of ECR* phase.

This phase can also be re-entered after a rollback of the type R1 or R2 (see 2.2.4). This phase does not involve comments related to cost impact in the event of fast-track execution of the type "estimation-based approval", see the normal flow description further below.

The *Commenting on ECR* phase starts when milestone M3.3, *ECR Detailed*, has been reached (see Figure 17). To start with, the ECR is prepared for evaluation in the subsequent steps in the phase *Prepare ECR for Comments*. The ECR is then evaluated in the *Give Comment on ECR* phase. This phase takes place in the *coordinator's* company or in partners' companies as well. Following this, the optional phase *Give Confirmation Comments* can be carried out in the company or in partners' companies. In this phase, a comment which has previously been submitted must be confirmed again, for example in the light of the other comments. The *Commenting on ECR* phase ends when milestone M3.4, *ECR Commented*, is reached.

Normal flow

The *Change Team* starts by preparing the ECR for evaluation by the Comment Performers in the activity *Prepare ECR for Comments* (see Figure 18). Following this, *Request_ECR_-comments* messages may be sent to the *Comment Performers*. It is possible that none or only one kind of comment message is requested (i.e. only one 'general' dialog). It is also possible to use several subdialogs for concurrent requests with each request dealing with a bundle of different kinds of comments. For example, it is possible to use one subdialog for technical comments only (which results in one message *Respond_ECR_comments* containing several comments with COMMENT.kind = 'development', 'prototype manufacturing',…) and one subdialog for review-evaluation comments (containing comments with COMMENT.kind = 'commercial').

***Note*:** *The subdialogs are distinguished by means of an attribute in the message header, see 3.3.4).*

Each Comment Performer processes the request message in the activity *Give Comment on ECR*, completing his work by the deadline specified in the request message. In doing so, he reviews the ECR and evaluates the change with respect to particular specialist topics and reports any evaluation data which was requested or agreed back to the *Change Team* in the response message.

Note: *If the deadline is not observed, the* Change Team *can contact the* Comment Performer *by phone, e-mail etc.*

The *Change Team* consolidates the comments in order to prepare the *Approval of ECR*, and the activity *Commenting on ECR* finishes in the status 'commented' documented in STATUS.status. Optionally, the The *Change Team* may send the message *Notify_ECR_commented*, indicating the completion of the phase *Commenting on ECR*.

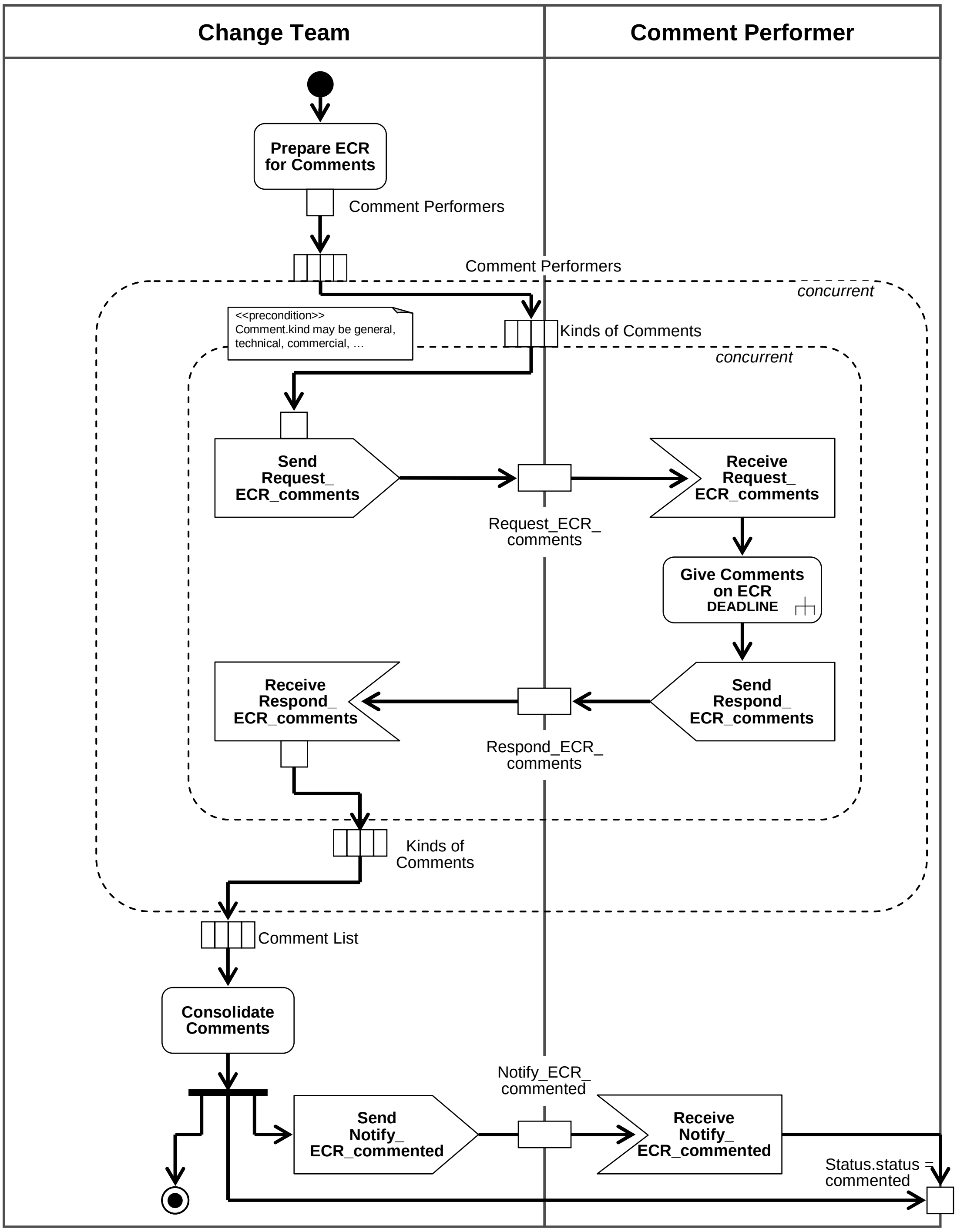

Figure 18: Normal flow for Commenting on ECR

Special flow variants and exceptions

In addition to this normal flow, two different flow variants may occur in the activity *Commenting on ECR* (see Figure 19 and Figure 20).

1. Process variant "Comments with Confirmation": If the process variant "Comments with confirmation" is agreed with the partner, the *Change Team* sends a *Request_ECR_confirmation_comment* message to the Comment Performers after all *Respond_ECR_-*

comment messages have been collected. With this message, the Comment Performers are requested to again confirm the validity of the comment previously submitted in the *Respond_ECR_comment* message. The *Request_ECR_confirmation_comment* message can contain other comments which must be taken into account when the Comment Performers confirm their own comment. In the activity *Give Confirmation Comment,* every Comment Performer requested to do so checks the validity of his comment by the deadline specified in the *Request_ECR_confirmation_comment* message and responds with a *Respond_ECR_confirmation_comment* message. If a *Respond_ECR_confirmation_comment* message does not contain a confirmation of the existing comment, the *Change Team* or the *EC Manager* can deal with this on a case-by-case basis, for example by rolling the ECR process back or by leaving the comment unconfirmed. The *Commenting on ECR* activity ends after all the response messages have been received by the *Change Team*.

2. Process variant *"Request / Respond Comment Message Repetition":* If the ECR Reference Process variant *"Request / Respond Comment Repetition"* is agreed with the partner, the *Change Team*, after receiving a *Respond_ECR_comment* message, can decide whether to repeat the request message/response message sequence with a Comment Performer in the *Decide Repetition* activity. (Note: If specific subdialogs are used, this can be used specifically for each subdialog.) A Comment Performer can express his wish to repeat the request message/response message sequence by sending a *Notify_ECR_comment_repetition_desired* message to the *Change Team*. After the *Decide Repetition* activity, the *Change Team* can restart the sequence at the point where the *Request_ECR_comment* message is sent if it is decided that the request message/response message sequence is to be repeated (continue branch in the diagram). If it is decided not to repeat the request message/response message sequence (else branch in the diagram), this variant ends.

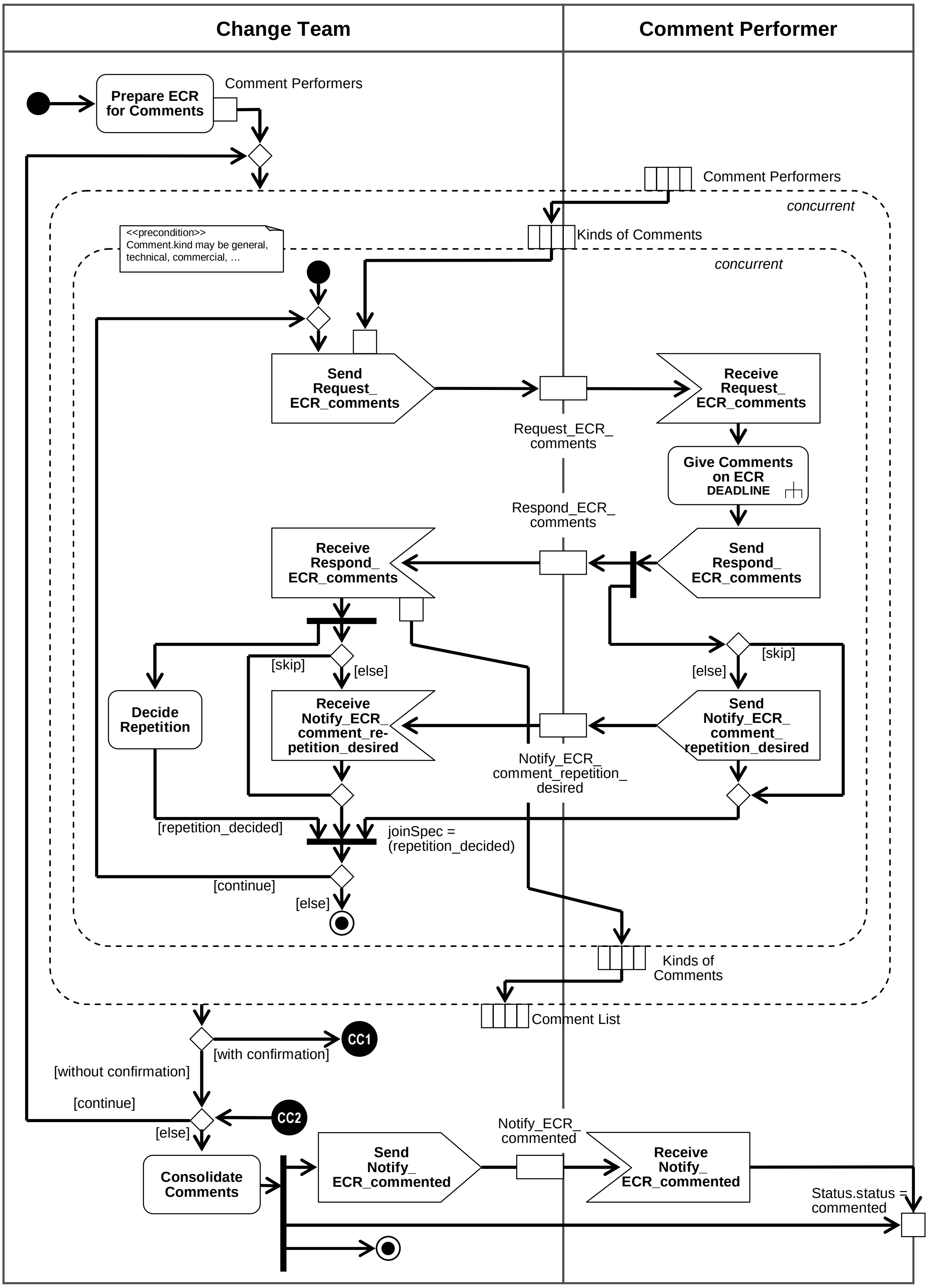

Figure 19: Flow variants for Commenting on ECR (1/2)

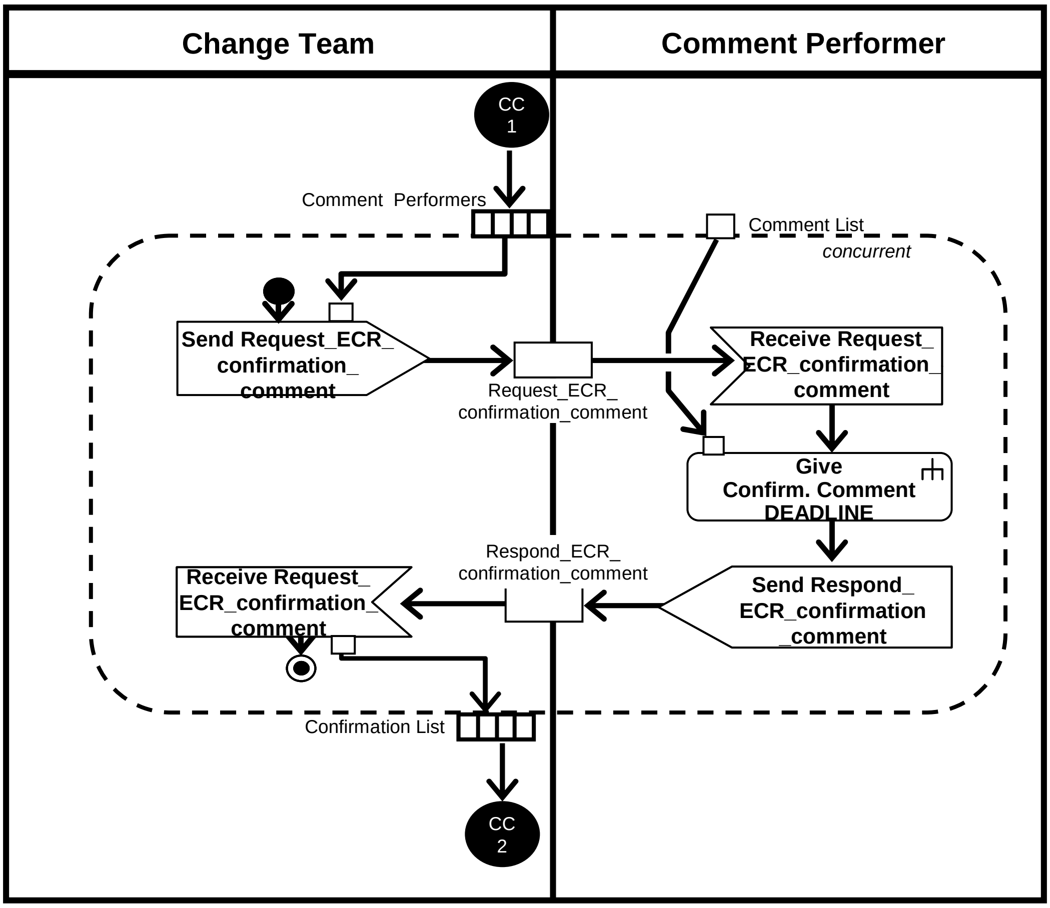

Figure 20: Flow variants for Commenting on ECR (2/2)

2.4.5 Approval of ECR phase

This section provides a detailed description of the *Approval of ECR* phase.

Purpose

The *Approval of ECR* phase starts after completion of the *Commenting on ECR* phase or, if the fast-track execution variant "estimation-based approval" has been selected, after completion of the *Technical Analysis of ECR* phase. After the ECR has been evaluated in the previous phase, a decision can be taken in this phase. Using the comments from the *Commenting on ECR* phase (or, in the same way, using the estimation values from the *Technical Analysis of ECR* phase), a decision is taken, justified and documented. To provide the basis for the coordinator's decision, it is possible to request beforehand that partners check the completeness of the ECR in the *Check Completeness of ECR* phase or to approve the ECR in the *Decide on ECR* phase. The result of this phase is that the ECR is approved or rejected and reasons for the decision are provided on the basis of the evaluation from the previous phase. It is also possible that the ECR may need to be revised. Typically, the decision team comprises senior managers, and the decision is made in a meeting without using the ECM system directly. The decision team can, however, also correspond to the *Change Team,* for example, depending on the expected costs of the ECR. The precise manner, in which a decision is taken, however, has no impact on the partners. This phase can also be re-entered after a rollback (see above).

Overview of phase

The *Approval of ECR* phase starts when the milestone M3.4, *ECR Commented*, is reached and begins with the optional phase *Prepare Approval Request* if external approval of the ECR by one or more partners is necessary (see Figure 21).

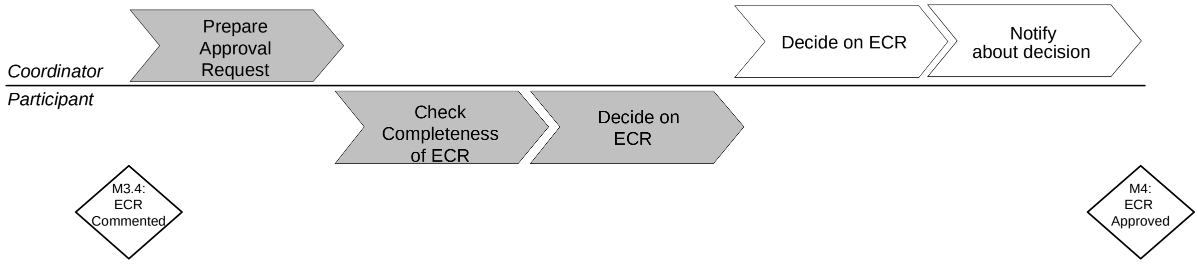

Figure 21: Phase overview for Approval of ECR

In this case, one or more partners can check the ECR for completeness in the next optional phase *Check Completeness of ECR*. If an ECR is incomplete, this can lead to a rollback (see above). If the ECR is complete, the *participants* provide the external approval or rejection in the next phase *Decide on ECR*. In all cases, the coordinator takes a decision on the ECR and its documentation in the next phase *Decide on ECR*. This is followed by the phase *Notify about Decision*, in which those affected by the ECR are informed about the decision. If the ECR was approved, milestone M4, *ECR Approved,* is reached. This completes the *Approval of ECR* phase.

Note**:** *After the* Approval of ECR *phase, the ECR will be implemented typically by one or more ECO's by modifying the product definition data according to the approved change specification. In addition, each of the finally released ECO's will be typically physically implemented in the product by one or more MCO's, e.g. in a plant. After the changes, specified by the ECR are implemented, the ECR may be closed by sending the Message* Notify_ECR_closed *(see also Figure 37), but this Message is* <u>*not*</u> *shown in the following process diagrams.*

Normal flow

At the beginning of this activity, the *Change Team* prepares the *coordinator's* decision in the activity *Prepare Decision* (see Figure 22, start at R4) by creating an ECR decision sheet from the evaluated ECR.

After this, the *Change Team* can modify the *Distribution List* in the activity *Modify Distribution List* in order to specify or modify the recipients of the decision notification. The *Decision Team* then makes a decision regarding the ECR and informs the *Change Team* of this decision. The *Change Team* documents the decision of the *Decision Team* in the ECM system in the activity *Document Decision on ECR*. Notification of the decision is then sent to the members of the *Distribution List* as a *Notify_ECR_decided* message. Depending on the result of the activity *Document Decision on ECR*, the *Approval of ECR* activity documents IN ACCEPTANCE.decision either that the ECR was approved ('accepted') or disapproved ('rejected'), or that a revision of the ECR was decided starting from Analysis of ECR (‘revision_needed_-back_to_analysis’), from Commenting on ECR (‘revision_needed_back_to_commenting’) or from Approval of ECR (‘revision_needed_back_to_approval’) (see 2.2.4).

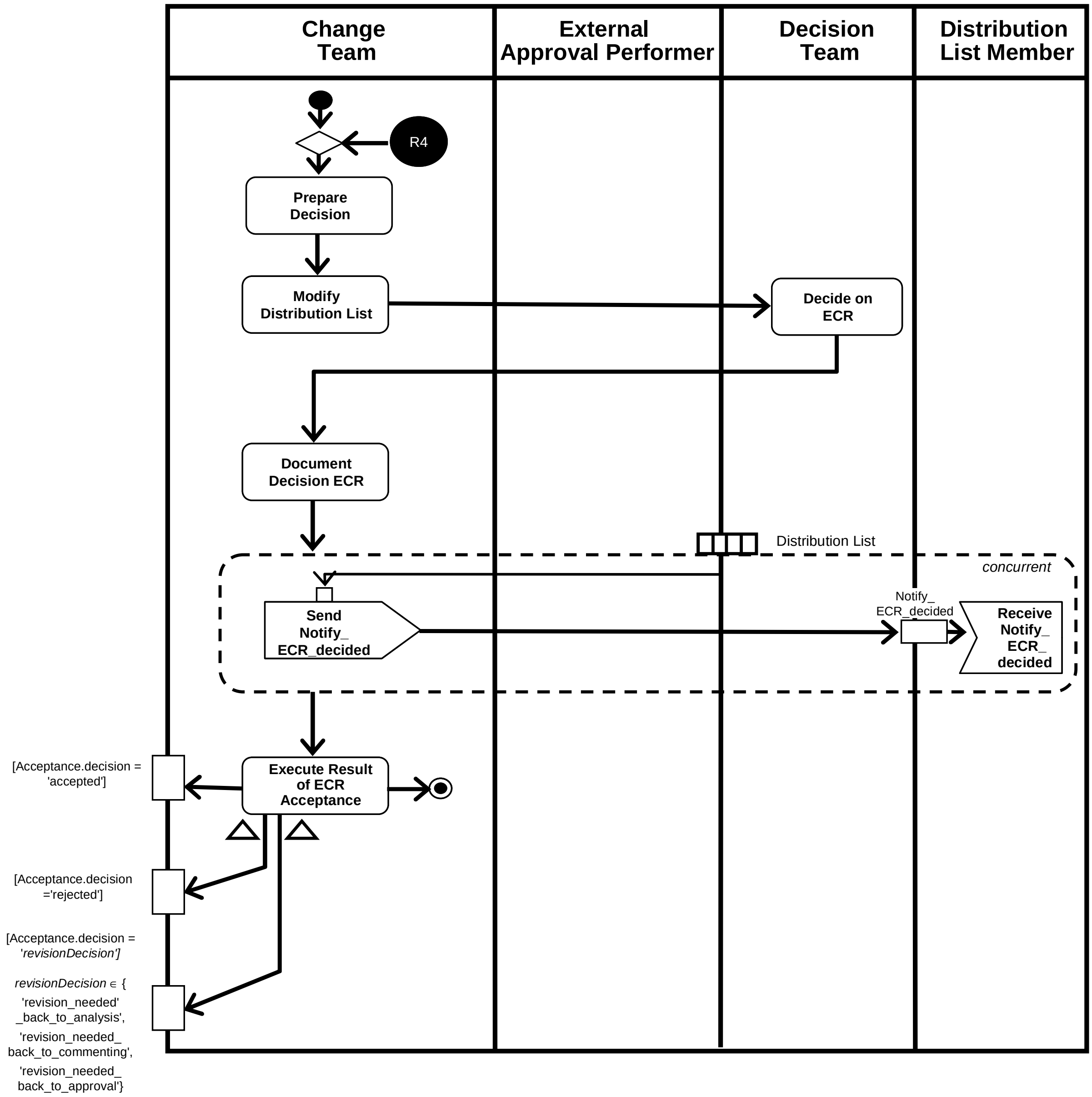

Figure 22: Normal flow for Approval of ECR

Special flow variants and exceptions

If external approval is required for a ECR, the activity *Approval of ECR* begins with appropriate interactions with one or more *External Approval Performers* (see Figure 23). The *Change Team* first selects those *participants* who are to be requested to give approval in the activity *Select Approval Performers*. If the "with Completeness Check" variant has been agreed, the *Change Team* prepares a *Request_ECR_completeness_check* message which is sent to an *External Approval Performer*. This is done in the *Prepare ECR Completeness Check Request* activity. The *External Approval Performer* receives the message and checks the ECR it contains in the *Check Completeness of ECR* activity to ensure that it is complete from his point of view, for example with respect to the scope of the change and the comments it contains.

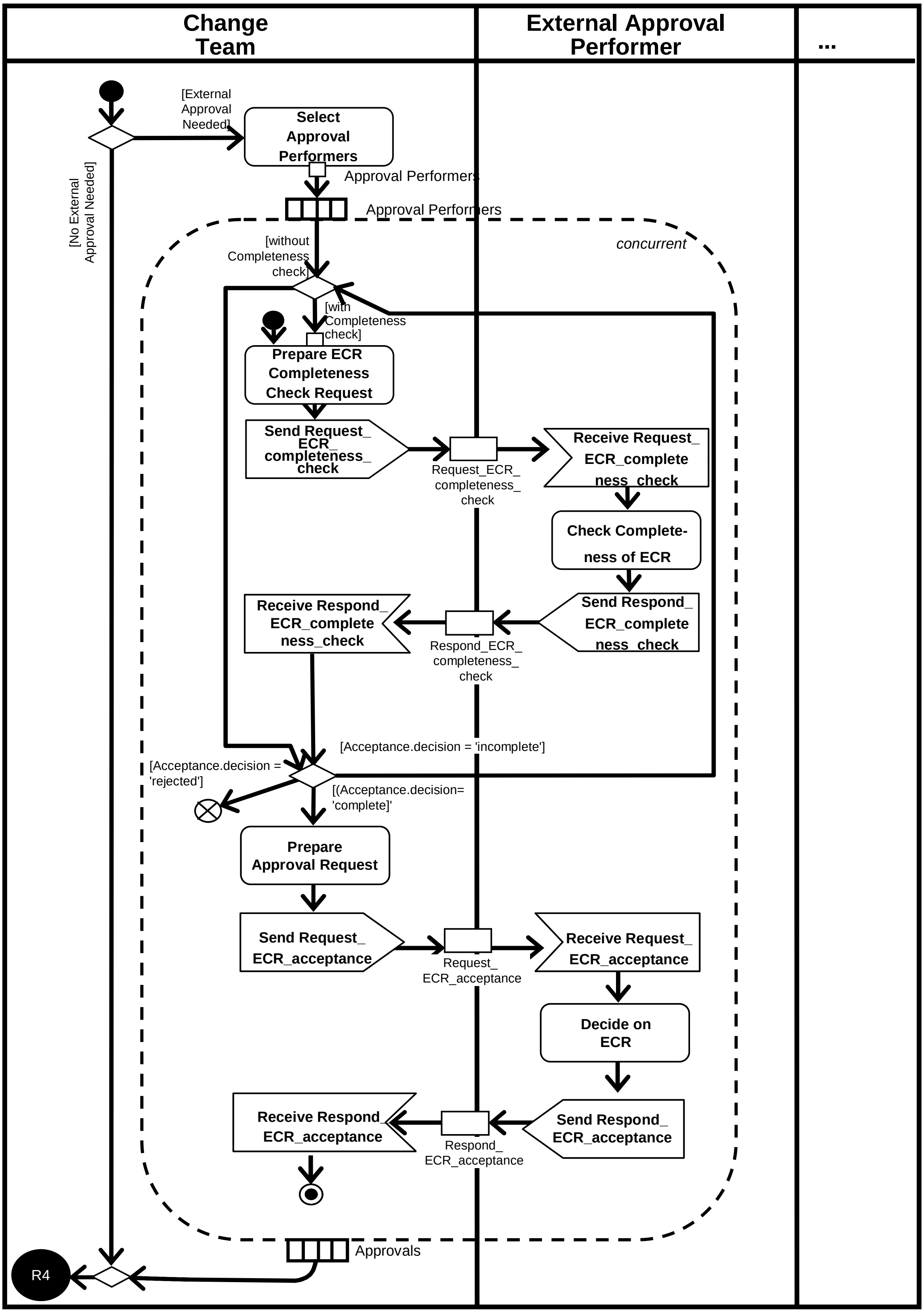

Figure 23: Flow variants for Approval of ECR

As the result of this activity, he sends the *Respond_ECR_completeness_check* message back to the *Change Team*,. If the *External Approval Performer* considered the ECR to be incomplete (ACCEPTANCE.decision = 'incomplete'), a further iteration of the loop described is carried out, i.e. the *Change Team* can prepare a new request in the *Prepare ECR Completeness Check Request* activity, etc. (If the response of the *External Approval Performer* necessitates a rollback, this can be carried out as described in 2.2.4.) The *External Approval Performer* may also reply that the ECR is rejected (ACCEPTANCE.decision = 'rejected'). The *External Approval Performer* may also reply that he considered the change request to be complete (ACCEPTANCE.decision = 'complete').

In this case, or if the *with Completeness Check* variant has not been agreed, the *Change Team* prepares a request for approval by an *External Approval Performer* in the activity *Prepare Approval Request* and as a result sends the *Request_ECR_acceptance* message. The *External Approval Performer* receives the message, decides on the ECR from his point of view and then returns his decision in the *Respond_ECR_acceptance* message. The *Change Team* then proceeds as described at the beginning of section 2.4.5.

3 ECR Interaction Scenarios and ECR Messages

Depending on the different business requirements and contractual conditions which apply, different data describing an ECR needs to be exchanged at different times in the different cooperative ventures. This section describes four typical ECR Interaction Scenarios (IS1, IS2, IS3, and IS4) that might be applied in a cross-company ECR process.

The following Figure depicts the intensity of interaction between the *coordinator* and the *participant* in the phases of the ECR Reference Process for each ECR Interaction Scenario.

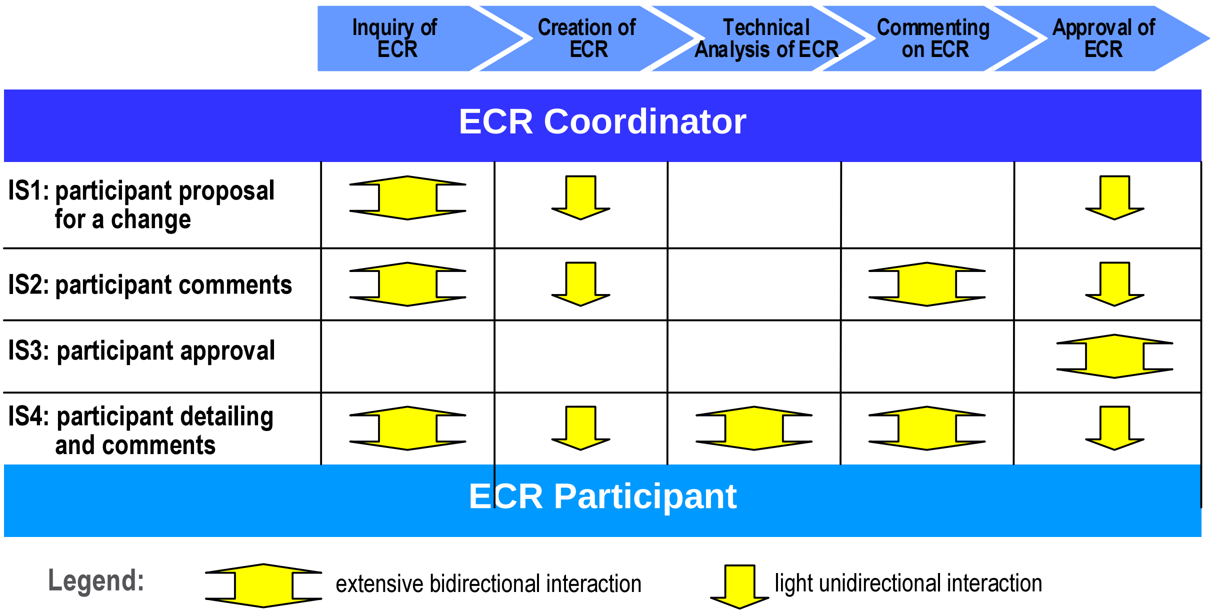

Figure 24: Interaction flow intensity in the typical ECR Interaction Scenarios

These four typical scenarios are:

- ***ECR Interaction Scenario 1:*** *Participant* proposal for a change.
  Within IS1, the *participant* is typically a supplier. He triggers the change but is not affected by the change itself. Therefore, the *participant* does not need much information about the ECR and is only loosely coupled with the *coordinator's* ECR process. ECR updates from the *coordinator*, rollbacks in the ECR Reference Process, and revisions of an ECR in the approval phase, if any, are *coordinator* internal only and are not visible for the *participant*.
- ***ECR Interaction Scenario 2:*** *Participant* comments.
  Within IS2, the *participant* is typically a supplier. The ECR process is driven by the *coordinator*. The *participant* may or may not trigger the change, but the *participant* in IS2 is always affected by the change itself; especially if the *participant* is involved in the commenting phase of the ECR Reference Process. Therefore the *participant* needs more information about the ECR and is more closely coupled with the *coordinator's* ECR process. ECR updates from the *coordinator*, rollbacks in the ECR Reference Process, and revision of an ECR in the approval phase which is visible for the *participant* may occur.
- ***ECR Interaction Scenario 3:*** *Participant* approval.
  Within IS3, the participant is typically a customer and the *coordinator* is typically a sup-

plier. The ECR process is driven by the *coordinator*. The communication in IS3 takes place in the approval phase of the ECR Reference Process only. Rollbacks in the ECR Reference Process and revisions of an ECR in the approval phase, if any, are *coordinator* internal only and are not visible for the *participant*.

- ***ECR Interaction Scenario 4:*** *Participant* detailing and comments.
  This ECR Interaction Scenario is similar to IS2 with the following exception: the participant is already involved in the technical phase of the ECR Reference Process. Therefore the *participant* needs more information about the ECR and is more closely coupled with the *coordinator's* ECR process. ECR updates from the *coordinator*, rollbacks in the ECR Reference Process, and revision of an ECR in the approval phase which is visible for the *participant* may occur.

The following table provides an overview of the role played by the *participant* in each ECR Interaction Scenario (Legend: "X": the *participant* plays that role; "O": the *participant* may play that role; "-": the *participant* does not play that role).

Table 1: Overview of the roles of the participant in ECR Interaction Scenarios

role of participant / Interaction Scenario	requestor	analysis performer	comment performer	external approval performer
Interaction Scenario 1	X	-	-	-
Interaction Scenario 2	O	-	X	-
Interaction Scenario 3	-	-	-	X
Interaction Scenario 4	O	X	O	-

3.1 Description method

An ECR Interaction Scenario describes how ECR data is exchanged between partners for a particular purpose using messages in order to implement process-oriented integration for a specific type of cooperation. ECR Interaction Scenarios make use of certain options for interaction provided by the ECR Reference Process in order to exchange the associated messages in a specific application context. A distinction is made between two different types of interaction: In unidirectional interactions, a single message of the type *Notify* is sent; in bidirectional interactions, a message of the type *Request* and one message of the type *Respond* are exchanged.

In the following sections, the ECR Interaction Scenarios are briefly described by illustrating the application context and showing the assignment of the tasks between the *coordinator* and the *participant* in the engineering change request process.

This is supplemented by a description (if necessary for each variant of this ECR Interaction Scenario) of the message flow involved in the exchange of ECR messages. A message flow diagram shows the possible flow of messages specific to the ECR Interaction Scenario from the *coordinator* to the *participant* and vice versa.

The ECR Messages are assigned to the phases of the ECR Reference Process of the *coordinator* in order to show the process context in which a ECR Message is exchanged. This process context is precisely defined in section 2.4 of this document. The sending and receiving actions of the *coordinator* are related with the sending and receiving actions the ECR Reference Process provides in general.

The meaning of the ECR Messages which can occur in different ECR Interaction Scenarios are described in sections 3.3.1 and 3.3.2.

Note: *Some ECR Messages (e.g. Notify_ECR_detailed, Notify_ECR_commented, Notify_-ECR_rolled_back_to_analysis, Notify_ECR_rolled_back_to_commenting, Notify_-ECR_closed) and other conditions which apply to the ECR Message sequences, in particular for the flow variants and exceptions, are described by the ECR Reference Process only and are not explicitly illustrated in the following ECR Interaction Scenario diagrams.*

3.2 ECR Interaction Scenarios

Note: *The ECR Interaction Scenario Protocol Specification in Appendix A describes the ECR Message flow explicitly and in more detail from the* coordinator's *and* participant's *point of view. It also includes all the relevant ECR Messages from the ECR Reference Process.*

3.2.1 ECR Interaction Scenario IS1 – Participant proposal for a change

Application context

This ECR Interaction Scenario is used if the *participant* identifies the need for a change but is not directly affected by the change itself (for example a change to a part which is the responsibility of another partner/supplier). The *participant* is therefore not involved in the subsequent processing of the ECR.

Typical application: Engineering changes to support a continuous improvement process (independent of a cooperation scenario); when an engineering partner is not affected by the issue for which he suggests an improvement measure.

In this ECR Interaction Scenario, an OEM typically plays the role of *coordinator* and a supplier typically plays the role of *participant*.

Message sequences

The following diagram provides an overview of the permitted sequences of messages in this ECR Interaction Scenario:

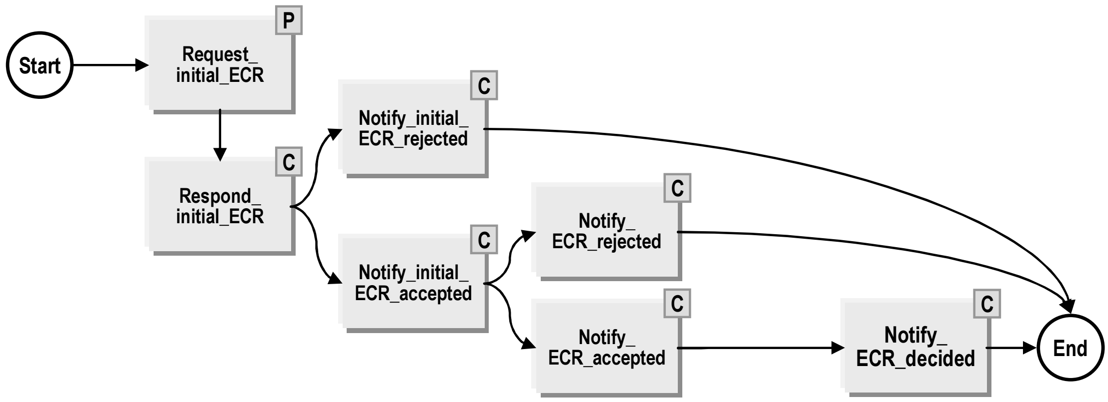

Figure 25: Permitted sequences of messages for participant proposal for a change (IS1)

The *participant* identifies the need for an ECR, i.e. he requests an initial ECR (*Request_-initial_ECR*) from the *coordinator*. The *participant* then receives confirmation that his request has been received under the initial ECR number of the *coordinator* (*Respond_initial_ECR*). Later on, the *coordinator* informs the *participant* whether the initial ECR was rejected (*Notify_initial_ECR_rejected)* or accepted *(Notify_initial_ECR_accepted*). In this case, a regular ECR was created based on the initial ECR. Later on in the creation phase, the *coordinator* informs the *participant* whether the regular ECR was rejected (*Notify_ECR_rejected)* or accepted (*Notify_ECR_accepted*) by the decision team. If the ECR was accepted, the *participant* finally receives a message notifying him whether the ECR was ultimately approved or rejected by the *coordinator* (*Notify_ECR_decided*) at the end of the engineering change request process.

The following diagram provides an overview of the messages in IS1 in the context of the ECR Reference Process. See section 3.3 for a description of the individual *messages*.

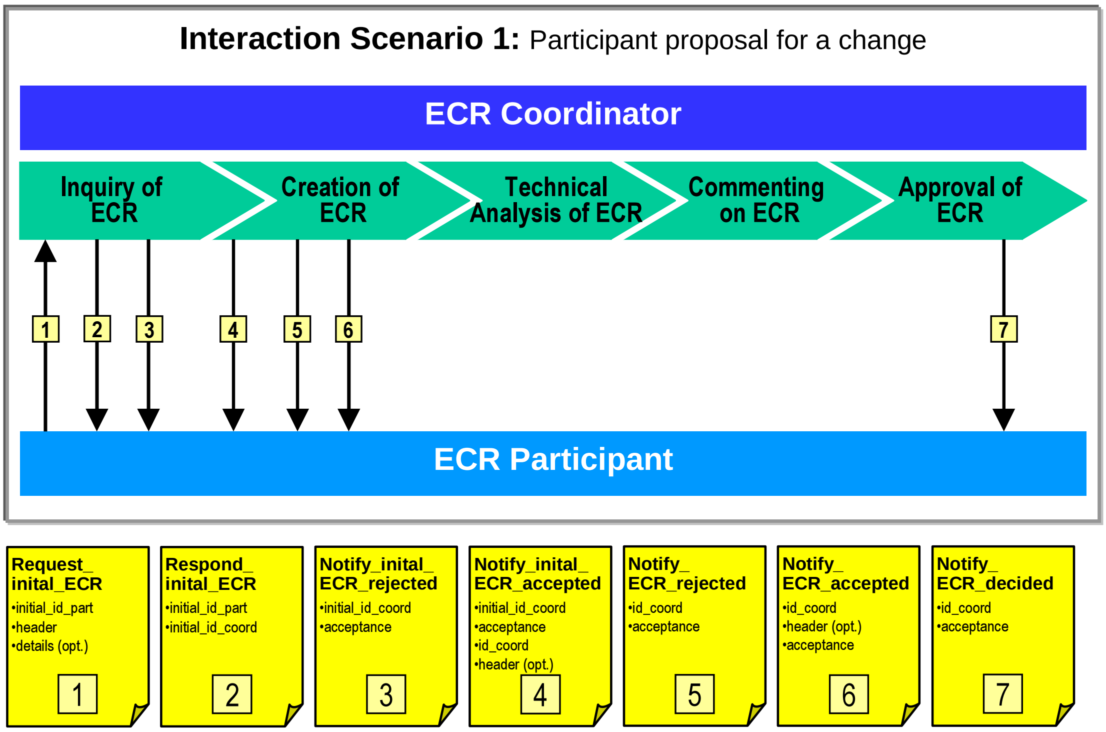

Figure 26: Messages in IS1 in the context of the ECR Reference Process

Note: *The parameters shown in the simplified signature of the messages (e.g. Id_partner in the message Request_initial_ECR) do not denote elements of the ECR data model but rather the name of the most important parameters of that message. The complete signature of the ECR Messages is provided in section3.3.*

3.2.2 ECR Interaction Scenario IS2 – Participant comments

Application context

In this ECR Interaction Scenario, the *participant* is requested to perform a technical and/or commercial evaluation and provide comments to the ECR of the *coordinator*. The *participant* was not involved in the preceding technical analysis and detailing phase.

Typical application: Production of parts by the *participant* for a vehicle of the *coordinator.*

Typically, the OEM takes on the role of *coordinator*. For parts developed and/or manufac-tured externally, a supplier plays the role of *participant* and provides comments for the parts which affect him or for which he is responsible.

Message sequences

The following diagram provides an overview of the permitted sequences of messages in this ECR Interaction Scenario:

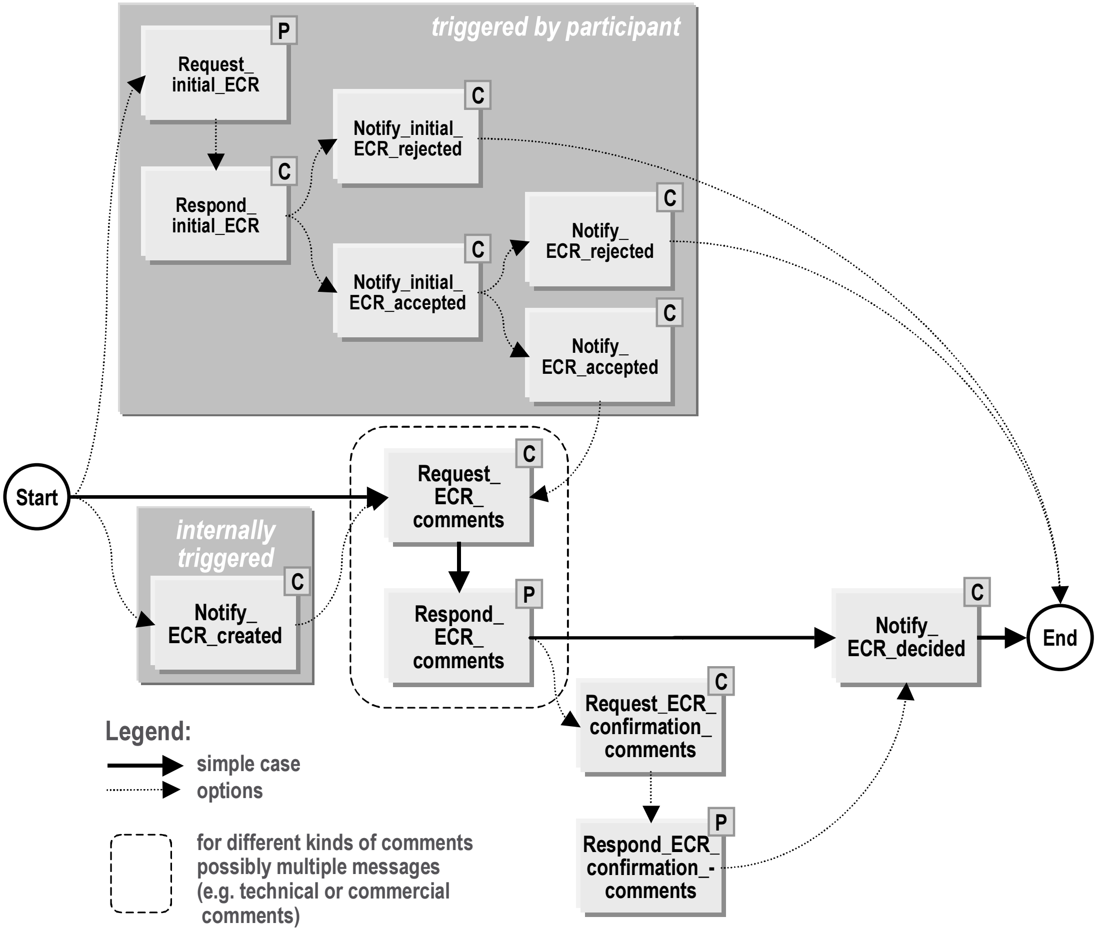

Figure 27: Permitted sequences of messages for participant comments (IS2)

Simple case:

The *coordinator* involves the *participant* in the commenting phase by requesting comments from the *participant* (*Request_ECR_comments*). The *participant* provides his comments on the ECR, generally including costs of the proposed change described by the ECR (*Respond_ECR_comments*). Finally, the *participant* receives a message notifying him whether the ECR was ultimately approved or disapproved by the *coordinator* (*Notify_ECR_decided*) at the end of the engineering change process.

Options:

a) In addition to the simple case, the *coordinator* can request that the *participant* provide final comments on the basis of a complete description of the ECR, if necessary (*Request_ECR_confirmation_comments, Respond_ECR_confirmation_comments*). This gives the *participant* the opportunity to confirm his own comments or, on occasion, to extend them or parts of them if the participant becomes aware of additional constraints as a result of the consolidated and detailed information and other in comments available in the meantime.

b) If an ECR was triggered at the *coordinator* side (as above), the *coordinator* may inform members of the *Distribution List* about the creation of the ECR (*Notify_ECR_created*) at an early stage. The subsequent procedure is the same as described for the simple case, possibly combined with option a).

c) If a *participant* triggers the ECR, he can request an initial ECR (*Request_initial_ECR*) at the*coordinator* side. The participant first receives confirmation that his request has been received under the initial ECR number of the coordinator (*Respond_initial_ECR*). Later on, the *coordinator* informs the *participant* as to whether the initial ECR has been rejected (*Notify_initial_ECR_rejected*) or accepted (*Notify_initial_ECR_accepted*). In this case, a regular ECR is created based on the initial ECR. Later on in the creation phase, the *coordinator* informs the participant whether the regular ECR was rejected (*Notify_ECR_rejected*) or accepted (*Notify_ECR_accepted*) by the decision team. If the ECR was accepted, the subsequent procedure is the same as described for the simple case, possibly combined with option a).

The following diagram provides an overview of the messages in IS2 in the context of the ECR Reference Process. For a description of the individual *messages*, see section 3.3.

Notes: *In IS2 no changes or additions to the results of the technical analysis phase should be made by the* participant *in the Respond_ECR_comments message. If the* participant *identifies the need to modify existing and/or to add new affected objects or properties, he should attach this information to the* ECR_comment *data object.*

In certain circumstances the Request_ECR_comments message (and therefore also the Respond_ECR_comments message) may not occur in IS2, e.g. if the commenting phase is performed using a procurement IT-system.

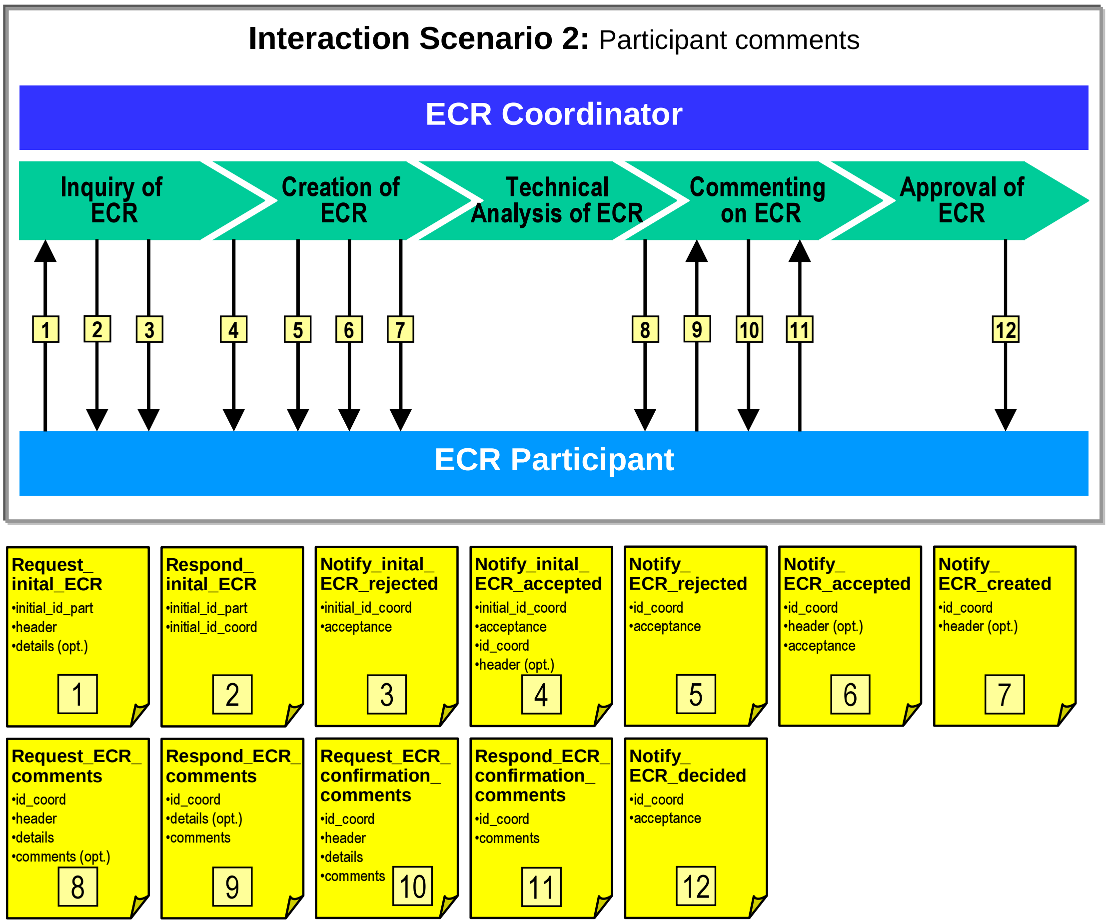

Figure 28: Messages in IS2 in the context of the ECR Reference Process

3.2.3 ECR Interaction Scenario IS3 – Participant approval

Application context

This ECR Interaction Scenario comes into play when the supplier has full responsibility for development of certain sets of parts of the OEM and these are to be changed on the initiative of the partner, or when the supplier acts as the general contractor in the vehicle project.

Typical application: Development and production of vehicles by a prime contractor for an OEM; when supplier has the responsibility for a part or assembly that is subject to an engineering change.

Typically, the prime contractor or supplier takes the role of the *coordinator* and the OEM takes on the role of the *participant*.

In this scenario, the need to make a change is identified only by the *coordinator*, and communication only takes place in phase 5 of the ECR Reference Process.

Message sequences

The following diagram provides an overview of the permitted sequences of messages in this ECR Interaction Scenario:

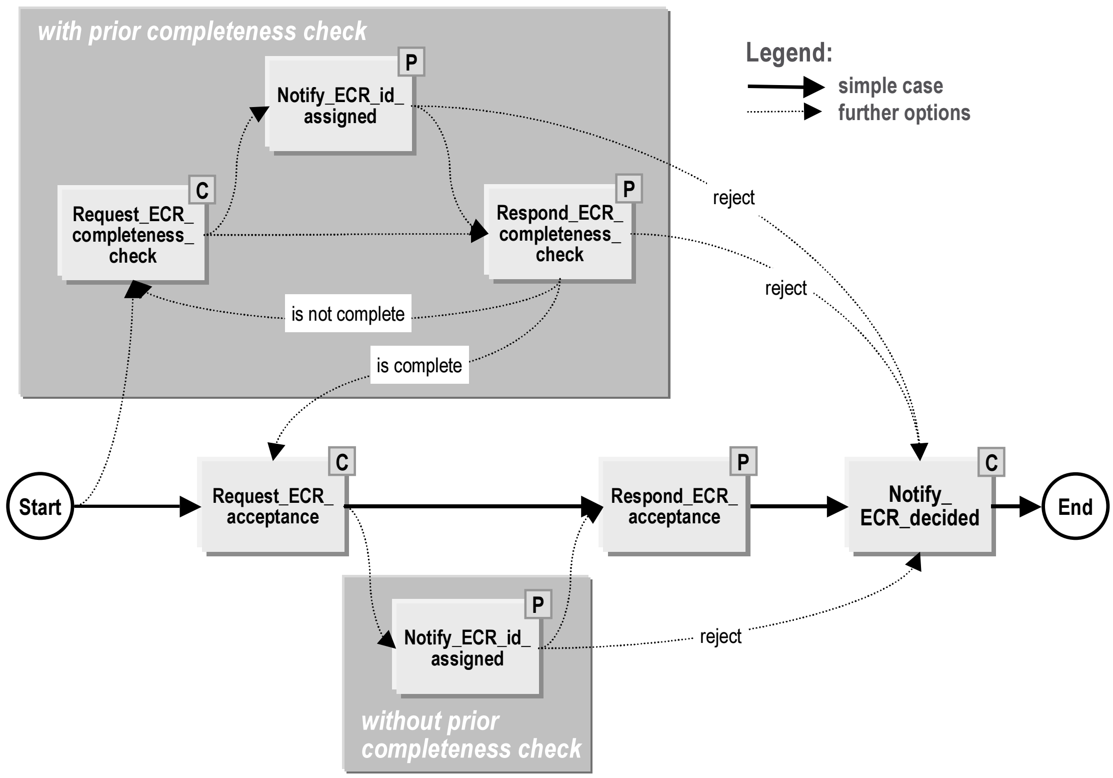

Figure 29: Permitted sequences of messages for participant approval (IS3)

Simple case:

The *coordinator* issues an ECR internally, describes the ECR in detail, makes comments on the proposed changes from his point of view and then sends this ECR, which is complete from his point of view, to the *participant*, requesting his approval (*Request_ECR_acceptance*). The *participant* analyzes the ECR and sends his response to the *coordinator* (*Respond_ECR_acceptance*). This response takes the form of approval or disapproval of the ECR from the point of view of the *participant*. Finally, the *coordinator* informs the *participant* of the final decision regarding the ECR (*Notify_ECR_decided*).

Notes: *The request can be approved by the* coordinator *even though it has been disapproved by the* participant*, for example by creating a new part rather than changing an existing part*

In the simple case, the communication is based on the ECR_ID of the coordinator*.*

Options:

a) In addition to the simple case, it can be agreed between both parties that the *coordinator* involve the *participant* early on in the upcoming ECR (branch depicted as "with prior completeness check"). In this case, the *coordinator* issues an ECR internally, describes the ECR in detail, makes comments on the proposed changes from his point of view and sends this ECR, which is complete from his point of view, to the *participant*, requesting him to check the technical completeness of the request (*Request_ECR_completeness_check, Respond_ECR_complete-ness_check*)

- If the *coordinator's* ECR is incomplete from the point of view of the *participant*, the *participant* informs the *coordinator* about this fact (branch depicted as "is_not_complete") and includes the set of parts which are missing in his opinion and/or his comments. Following the *participant's* evaluation, the *coordinator* sends a further *Request_ECR_completeness_check* message.
- If, on the other hand, the ECR is complete, the *participant* informs the *coordinator* about it (branch depicted as "is_complete"). The *coordinator* then sends a *Request_ECR_acceptance* message.
- The *participant* can reject the request at this early stage (branch depicted as "reject"). The *coordinator* responds by sending a *Notify_ECR_decided* message.

As soon as the *coordinator* has received a positive response from the *participant* with respect to the completeness of the ECR, he can send the complete ECR to the *participant* using the *Request_ECR_acceptance* message, requesting that the *participant* approves the ECR.

b) In addition to the simple case or in addition to option a), it can be agreed between both parties that communication be based on the ECR_id of the *participant*. In this case the message *Notify_ECR_id_assigned* is sent once after the first message of the *coordinator*, i.e. after *Request_ECR_acceptance* in the simple case or after *Request_ECR_completeness_check* in option a). Subsequent communication is based on the *ECR_id* of the *participant*.
The message *Notify_ECR_id_assigned* contains the *ECR_id* counterparts at *participant* and *coordinator* side. The *participant* can also reject the ECR using this message (depicted case “reject”). This is done using the "reject" flag. In this case, the *coordinator* responds by sending a *Notify_ECR_decided* message.

The following diagram provides an overview of the messages in IS3 in the context of the ECR Reference Process. For a description of the individual *messages*, see section 3.3.

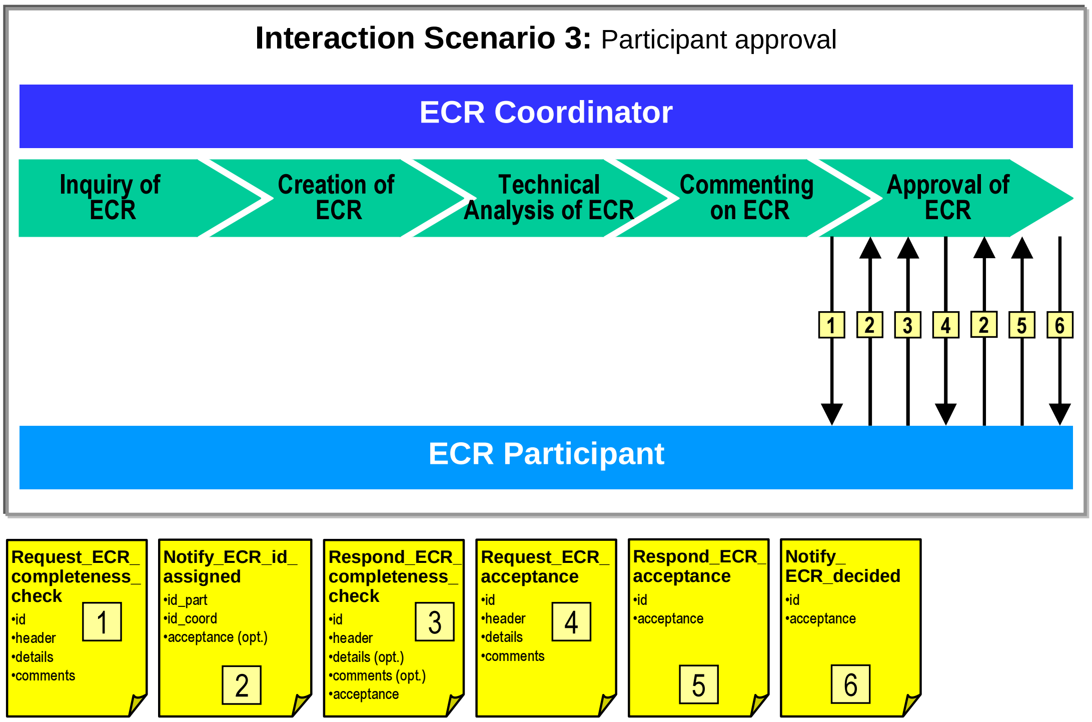

Figure 30: Messages in IS3 in the context of the ECR Reference Process

3.2.4 ECR Interaction Scenario IS4 – Participant detailing and comments

Application context

Interaction Scenario 4 describes a joint engineering change process in which the *coordinator* and the *participant* work closely together throughout all phases of that process.

Typical application: Development and production of entire vehicles in a cooperative venture.

Typically, the OEM takes the role of the *coordinator* and each of its cooperation partners takes the role of the *participant*.

Message sequences

The following diagram provides an overview of the permitted sequences of messages in this ECR Interaction Scenario:

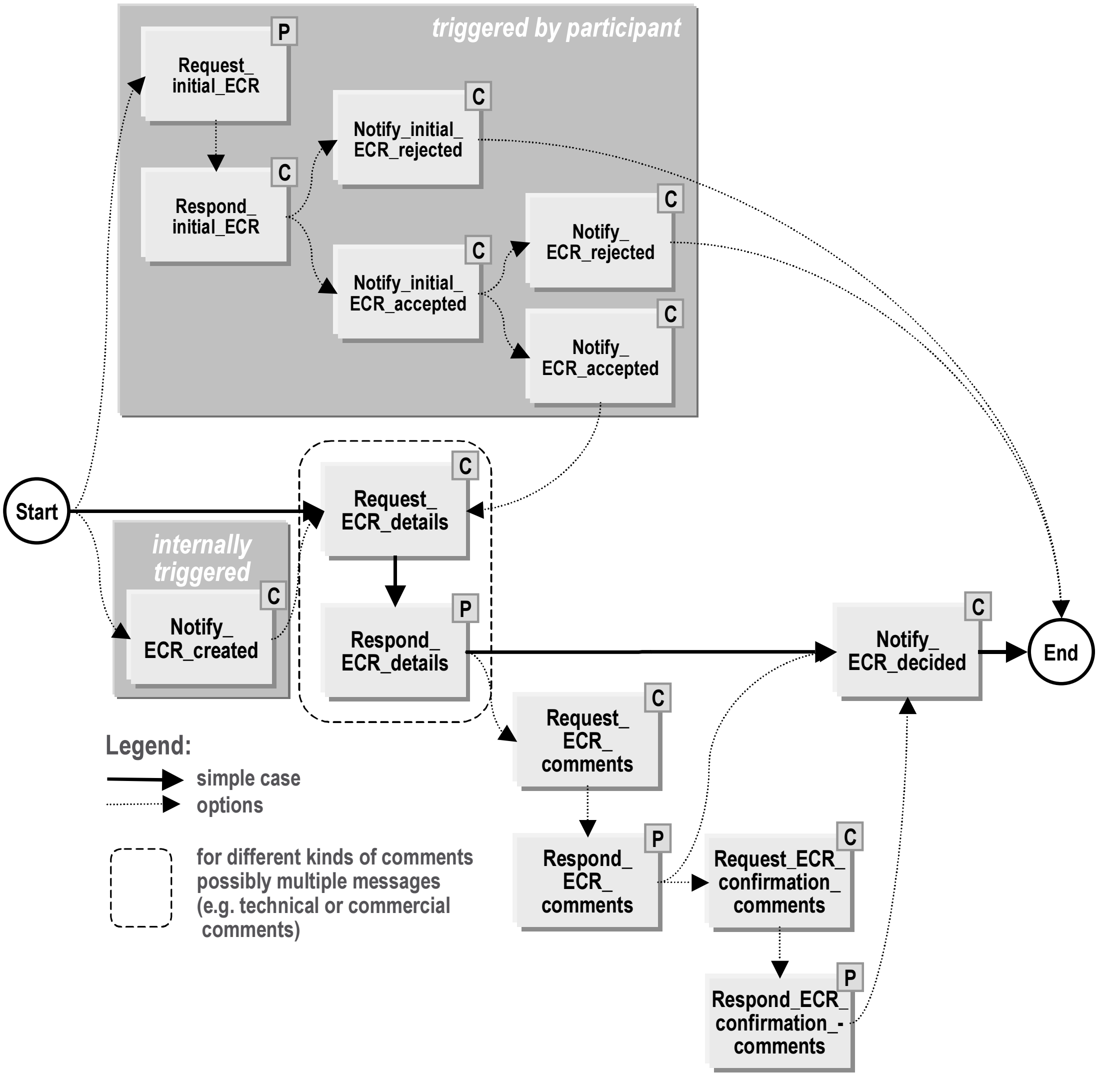

Figure 31: *Permitted sequences of messages for participant detailing and comments (IS4)*

Simple case:

The *coordinator* involves the *participant* in the technical analysis and detailing phase by requesting technical details from the *participant* (*Request_ECR_details*). The *participant* provides additional technical details of the ECR, generally including additional objects affected (e.g. parts), costs of the proposed change from a technical point of view and/or information relating to weight changes caused by the proposed ECR (*Respond_ECR_details*). Finally, the *participant* receives a message notifying him whether the ECR was ultimately approved or rejected by the *coordinator* (*Notify_ECR_decided*) at the end of the common engineering change request process.

Options:

a) In addition to the simple case, the *coordinator* can also involve the *participant* in the commenting phase, if required. He can request comments on individual affected objects listed in the ECR (e.g. the costs for the change of a certain part) or on the entire ECR including any comments already received from other *participants* (*Re-*

quest_ECR_comments, Respond_ECR_comments).
In addition, the *coordinator* can request that the *participant* provide final comments on the basis of a complete description of the ECR, if necessary (*Request_ECR_-confirmation_comments, Respond_ECR_confirmation_comments*) This gives the *participant* the opportunity to confirm his own comments or, on occasion, to extend them or parts of them if the *participant* becomes aware of additional constraints as a result of the consolidated and detailed information and other in the meantime available comments.

b) If an ECR was triggered at the *coordinator* side, the *coordinator* may inform members of the *Distribution List* about the creation of that ECR (Notify_ECR_created) at an early stage. The subsequent procedure is the same as described for the simple case, possibly combined with option a).

c) If a *participant* triggers the ECR, he can request an initial ECR (*Request_initial_ECR*) at *coordinator* side. The *participant* first receives a confirmation that his request has been received under the initial ECR number of the *coordinator* (*Respond_initial_ECR*). Later on, the *coordinator* informs the *participant* whether the initial ECR was rejected (*Notify_initial_ECR_rejected*) or accepted (*Notify_initial_ECR_accepted*). In this case, a regular ECR is created based on the initial ECR. Later on in the creation phase, the *coordinator* informs the *participant* whether the regular ECR was rejected (*Notify_ECR_rejected*) or accepted (*Notify_ECR_accepted*) by the decision team. If the ECR was accepted, the subsequent procedure is the same as described for the simple case, possibly combined with option a).

The following diagram provides an overview of the messages in IS4 in the context of the ECR Reference Process. For a description of the individual *messages*, see section 3.3.

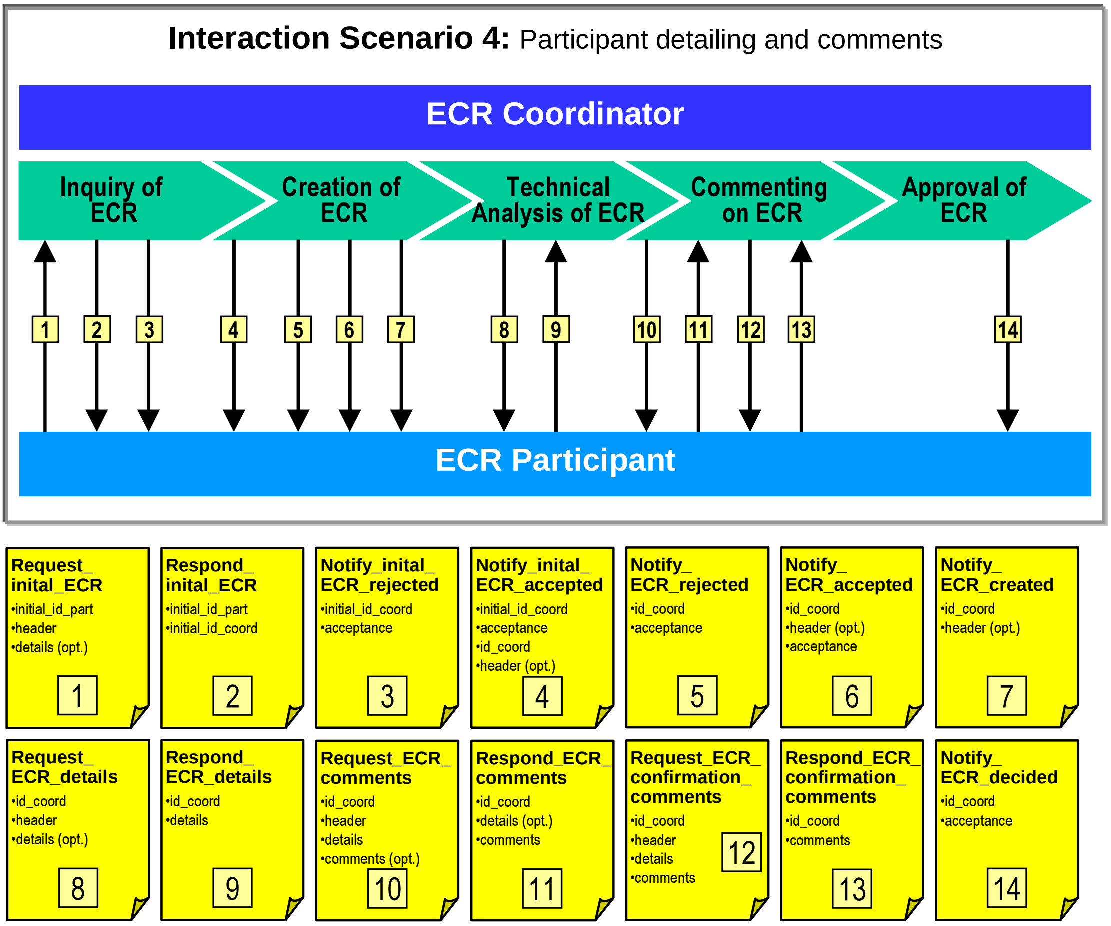

Figure 32: Messages in IS4 in the context of the ECR Reference Process

Note: *In IS4 no changes or additions to the results of the technical analysis phase should be made by the participant in the Respond_ECR_comments message. If the participant identifies the need to modify existing and/or to add new affected objects or properties, he should attach this information to the Comment data object.*

3.3 ECR Messages

In the following sections, the ECR Messages that can occur during the ECR Reference Process phases are described by explaining the semantics of the ECR Message as seen from an application-oriented point of view.

3.3.1 Data-oriented ECR Messages

To support data-oriented scenarios, common messages are defined which can be used in each of the ECR Reference Process phases. If data-oriented communication is used in combination with process-oriented communication, these messages can occur within the ECR Interaction Scenarios, but they are not described in the ECR Interaction Scenarios (section 3.2) itself.

Request_all_ECR_ids

This message from the *participant* requests all the existing identifying objects of an ECR in which the *participant* is involved and associated header information from the *coordinator*. This message itself does not contain any parameters.

Respond_all_ECR_ids

This message from the *coordinator* is the response to the *Request_all_ECR_ids* message. It contains all the ECRs including attached header information (either regular or initial header) in which the *participant* is involved.

Parameters:

- identifier of the ECR [mandatory]
- either the header information of the regular ECR associated with the ECR [optional], or
- the header information of the initial ECR associated to ECR [optional]

Notes: *The criteria for determining the ECRs and associated header information sent back to the* participant *are implementation specific.*

The result returned by this message may consist of a zero, one or more tuples containing ECR identification and associated header information.

Request_ECR_data

This message from the *participant* requests all the existing data objects of the data model for the given identifying information of an ECR (including a value for each of its mandatory attributes) from the *coordinator*.

Parameters:

- identifier of the ECR [mandatory]

Respond_ECR_data

This message from the *coordinator* is issued in response to the *Request_ECR_data* message. It contains all the data that are associated to the identifying object (ECR) of the previously received *Request_ECR_data* message.

Parameters:

- identifier of the ECR [mandatory]
- general detail information associated with ECR [optional] and/or
- specific detail information associated with ECR [optional] and/or
- comment information associated with ECR [optional] and/or
- acceptance information associated with ECR [optional]

Notes: *The result returned by this message must contain one or more of the optional parameters. The criteria for determining the parameters sent back to the* participant *are implementation specific.*

3.3.2 Process-oriented ECR Messages

The following describes all the messages permitted in the ECR Reference Process. There are two categories of ECR Messages:

- Messages that are described in the ECR Interaction Scenarios
- Messages that can occur in the ECR Interaction Scenarios, but which are not described in the ECR Interaction Scenario (section 3.2) itself.

Request_initial_ECR

This message from the *participant* contains a request to create an initial ECR.

Parameters:

- identifier of the initial ECR of the *participant* [mandatory]
- the header containing a description of the initial ECR [mandatory]

if available, the *participant* may include already existing details (e.g. affected parts) of the initial ECR [optional]

Respond_initial_ECR

This message from the *coordinator* is issued in response to the *Request_initial_ECR* message and confirms receipt of the initial ECR. In confirmation, the *coordinator* responds with his identifier of the initial ECR which is used for the further processing of the initial ECR.

Parameters:

- identifier of the initial ECR of the *participant* [mandatory]
- identifier of the initial ECR of the *coordinator* [mandatory]

Notify_initial_ECR_rejected

This message from the *coordinator* informs the *participant* that the initial ECR has been rejected during the phase inquiry of ECR.

Parameters:

- identifier of the rejected initial ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is always “pending”
- current status of the ECR [mandatory]
  Note: this state is always: “disapproved”
- the signature of the person responsible for the state change of the ECR, i.e. the signature of the acceptor and the reason for rejection of the initial ECR [mandatory]

Notify_initial_ECR_accepted

This message from the *coordinator* informs the *participant* that his initial ECR has been accepted and a regular ECR has been created based on the initial ECR during the phase inquiry of ECR. Identifier of the regular ECR is used for further processing of the regular ECR.

Parameters:

- identifier of the accepted initial ECR of the *coordinator* [mandatory]
- identifier of the regular ECR of the *coordinator* that has been created on basis of the initial ECR [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is always “pending”
- current status of the ECR [mandatory]
  Note: this state is always: “initiated”
- the signature of the person responsible for the state change of the ECR, i.e. the signature of the acceptor of the initial ECR [mandatory]
- the header containing a description of the regular ECR from the *coordinator* [optional]

Notify_ECR_rejected

This message from the *coordinator* informs the *participant* that the regular ECR has been rejected during the phase creation of ECR

Parameters:

- identifier of the rejected regular ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is always “initiated”
- current status of the ECR [mandatory]
  Note: this state is always: “disapproved”
- the signature of the person responsible for the state change of the ECR, i.e. the signature of the acceptor and the reason for rejection of the regular ECR [mandatory]
- the signature of the acceptor and the reason for rejection of the regular ECR [mandatory]

Notify_ECR_accepted

This message from the *coordinator* informs the *participant* that the previously created regular ECR has been accepted during the phase creation of ECR.

Parameters:

- identifier of the accepted regular ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is always “initiated”
- current status of the ECR [mandatory]
  Note: this state is always: “created”
- the signature of the person responsible for the state change of the ECR [mandatory]
- the header containing a description of the regular ECR from the *coordinator* [optional]
- the signature of the acceptor of the initial ECR [mandatory]

Notify_ECR_created

This message informs the *participant* that an ECR (relevant to the *participant*) has been created in the *coordinator's* system during the phase creation of ECR.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is always “initiated”
- current status of the ECR [mandatory]
  Note: this state is always: “created”
- the signature of the person responsible for the state change of the ECR [optional]
- the header containing a description of the ECR from the *coordinator* [optional]

Notify_ECR_detailed

This message from the *coordinator* informs the *participant* that the regular ECR has been detailed during the phase technical analysis of ECR is finished.

Parameters:

- identifier of the regular ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is either “created”, “detailed” or “commented”
- current status of the ECR [mandatory]
  Note: this state is always: “detailed”
- the signature of the person responsible for the state change of the regular ECR [optional]

Notify_ECR_commented

This message from the *coordinator* informs the *participant* that the regular ECR has been commented during the phase commenting on ECR is finished.

Parameters:

- identifier of the regular ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is either: “detailed” or “commented”
- current status of the ECR [mandatory]
  Note: this state is always: “commented”
- the signature of the person responsible for the state change of the ECR [optional]

Notify_ECR_closed

This message from the *coordinator* informs the *participant* that the ECR has been closed.

Parameters:

- identifier of the regular ECR of the *coordinator* [mandatory]

- pervious status of the ECR [mandatory]
  Note: this state is either: “approved”, “disapproved” or “canceled”
- current status of the ECR [mandatory]
  Note: this state is always: “closed”
- the signature of the person responsible for the state change of the ECR [optional]

Request_ECR_details

This message from the *coordinator* requests technical comments on the ECR or subsets of it from the *participant*. These technical details must include affected parts and their usage in platforms, series, model types and/or modules, estimated costs of the change from an engineering point of view as well as property and weight changes. Any existing information which the *participant* may require for a meaningful technical analysis of the change, such as the scope of the changes known to date, expected changes to costs and weight and any affected platforms, series, model types and/or modules, may be made available to the *participant*.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- the header containing a description of the ECR of the *coordinator* [mandatory]
- if available, the *coordinator* may include already existing details (e.g. affected parts) of the ECR [optional]

Respond_ECR_details

This message from the *participant* contains his technical comments on the ECR in response to a *Request_ECR_details* message.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- the details given by the *participant* as the result of his technical analysis of the change [mandatory]

Request_ECR_comments

This message from the *coordinator* requests comments on the ECR or subsets of it from the *participant*. Any existing information which the *participant* may require in order to provide meaningful comments on the change, such as the scope of the changes known to date, expected changes to costs and weight and any affected platforms, series, model types and/or modules, must be made available for the *participant*. Any existing comments may also be provided to the *participant*. This message may be used as a general request for comments on the ECR or as a dedicated request for a specific kind of requested comments, e.g. technical or review-evaluation comments.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- the header containing a description of the ECR of the *coordinator* [mandatory]

- the consolidated technical details (e.g. affected parts) of the ECR [mandatory]
- if available, the *coordinator* may include existing comments on the ECR [optional]

Respond_ECR_comments

This message from the *participant* contains his non-technical comments on the ECR in response to a *Request_ECR_comments* message. A non-technical comment may refer to the ECR in general as well as to certain subsets of it. Depending on the request message it is referring to, this message may be a response containing general comments on the ECR or containing a specific kind of comments, e.g. technical or review-evaluation comments.

Depending on the nature and scope of the change represented by the ECR, additional technical comments / details (e.g. affected parts) may also be included in this message.

Note: *In this case, the* ECR_comment *(and not the* ECR_detail*) object of the ECR requirements data model is used to carry the additional information. No new ECR_detail objects are to be attached to the ECR.*

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- if a comment refers to a subset of the ECR, this subset is represented by this parameter [optional]
- the comments of the *participant* as a result of the commenting phase for the change [mandatory]

Request_ECR_confirmation_comments

This message from the *coordinator* requests a final non-technical comment from the *participant* on the basis of a holistic description of the ECR including its header data, the consolidated details of the ECR and the consolidated comments which have been received from all involved parties. This request gives the *participant* the opportunity to confirm his own comments or, on occasion, to correct or withdraw them if the *participant* becomes aware of changed constraints as a result of the holistic view on the ECR.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- the header information for the ECR [mandatory]
- consolidated details of the ECR [mandatory]
- consolidated comments on the ECR [mandatory]

Respond_ECR_confirmation_comments

This message from the *participant* contains his final comments on the ECR in response to a *Request_ECR_confirmation_comments* message.

In individual cases, additional comments may also be included in this message.

Note: *In this case, the* ECR_comment *(and not the* ECR_detail*) object of the ECR requirements data model is used to carry the additional information. No new ECR_detail objects are to be attached to the ECR.*

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- final comments on the ECR from the *participant* [mandatory]

Notify_ECR_id_assigned

Note: *This message is used only if an agreement was made between the* participant *and the coordinator that the ECR will be processed using the* participant's *identifier of the ECR.*

This message from the participant *confirms receipt of the first Request_ECR_completeness_check message (if it is agreed that the ECR be processed according to the prior completeness check mechanism) or the first Request_ECR_acceptance message (without prior completeness check).*

If the ECR previously sent by the coordinator *is not acceptable from the point of view of the* participant*, the* participant *can reject the ECR in this message and notify the* coordinator *of this rejection.*

If, on the other hand, the ECR is acceptable from the point of view of the participant*, the* participant *notifies the* coordinator *of his own identifier of the ECR which is used for the further processing of the ECR.*

Parameters:

- identifier of the initial ECR of the *participant* [mandatory]
- identifier of the initial ECR of the *coordinator* [mandatory]
- the signature of the acceptor and the reason for rejection of the regular ECR [mandatory, if *participant* rejects the ECR, parameter not applicable otherwise]

Request_ECR_acceptance

This message from the *coordinator* requests an approval from the *participant* on the basis of a holistic description of the ECR including its header data, the consolidated details of the ECR and the consolidated comments.

Parameters:

- identifier of the ECR [mandatory]
- the header information for the ECR [mandatory]
- consolidated technical comments on the ECR [mandatory]
- consolidated non-technical comments on the ECR [mandatory]

Respond_ECR_acceptance

This message from the *participant* contains his binding response to the *Request_ECR_-acceptance* message.

Parameters:

- identifier of the ECR [mandatory]
- the signature of the acceptor and, if rejected, the reason for rejection of the ECR [mandatory]

Request_ECR_completeness_check

Note: *This message is used only, if an agreement was made between the* participant *and the* coordinator *that the ECR will be processed using prior completeness check mechanism, i.e. to involve the* participant *during the process of completing the ECR.*

This message of the coordinator *requests the* participant *to check the completeness of the ECR.*

Parameters:

- identifier of the ECR [mandatory]
- the header information for the ECR [mandatory]
- if available, consolidated details of the ECR [optional]
- if available, consolidated comments on the ECR [optional]

Respond_ECR_completeness_check

Note: *This message is used only if an agreement was made between the* participant *and the* coordinator, *that the ECR will be processed using a prior completeness check mechanism, i.e. involve the* participant *in the process of completing the ECR.*

This message from the participant *gives the* participant *an opportunity to check the ECR previously submitted by the* coordinator *with respect to technical completeness, supplement it if necessary or indicate an early rejection.*

If the ECR previously sent by the coordinator *is incomplete from the point of view of the* participant*, the* participant *can add new or modify existing technical details and/or comments.*

If, on the other hand, the ECR previously sent by the coordinator is complete from the point of view of the participant*, he notifies the* coordinator *of this.*

If the ECR previously sent by the coordinator *is not acceptable from the point of view of the* participant*, the* participant *can reject the ECR in this message and notify the* coordinator *of this rejection.*

Parameters:

- identifier of the ECR [mandatory]
  - if is_not_complete => expecting additional or modified header information of the ECR [optional]
  - if is_not_complete => expecting additional or modified details of the ECR [optional]
  - if is_not_complete => expecting additional or modified comments for the ECR [optional]

- the signature of the acceptor and, if rejected, the reason for rejection of the ECR [mandatory]

Notify_ECR_decided

This message from the *coordinator* is binding and notifies the *participant* of the decision regarding the ECR.

Parameters:

- identifier of the ECR [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is always: “commented”
- current status of the ECR [mandatory]
  Note: this state is either “approved” or “disapproved”
- the signature of the acceptor and, if rejected, the reason for rejection of the ECR [mandatory]

***Note*:** *The following applies in Interaction Scenario 3 only: even if the* participant *has previously rejected the ECR in the Respond_ECR_acceptance message, the ECR can still be approved by the* coordinator *(for example if a new part is created instead of an existing part being changed).*

The following messages are defined in the ECR Reference Process only; they are not described in the ECR Interaction Scenarios but can be applied in the all or certain Interactions Scenarios.

Notify_ECR_rolled_back_to_analysis

This *coordinator* message informs the *participant* that the ECR Reference Process was rolled back from the current phase (*Technical Analysis of ECR, Commenting on ECR or Approval of ECR*) to the start of the phase *Technical Analysis of ECR* for the *coordinator* ECR.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]

Notify_ECR_rolled_back_to_commenting

This *coordinator* message informs the *participant* that the ECR Reference Process was rolled back from the current phase (*Commenting on ECR or Approval of ECR*) to the start of the phase *Commenting on ECR* for the *coordinator* ECR.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]

Notify_ECR_canceled

This *coordinator* message informs the *participant* that the ECR Reference Process was canceled for the *coordinator* ECR.

***Note*:** *This message can be sent in the phases* Creation of ECR, Technical Analysis of ECR, Commenting on ECR or Approval of ECR *only.*

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- pervious status of the ECR [mandatory]
  Note: this state is either “created”, “detailed” or “commented”
- current status of the ECR [mandatory]
  Note: this state is either “approved” or “disapproved”

Notify_response_deferred

This optional message of the recipient of a previously received Request Message (e.g. Request_ECR_comments) informs the sender of that Request Message that the Respond Message to the original Request Message will be deferred.

Parameters:

- planed date of the delivery of the Respond Message [optional]

Notify_ECR_update

This *coordinator* message informs the *participant* that the *coordinator* ECR was updated. All relevant data of the updated ECR is attached to that message.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]
- the header information for the ECR [optional]
- if available, the consolidated details of the ECR [optional]
- if available, the consolidated comments on the ECR [optional]

Note: *At least one of the optional parameters must be provided.*

Notify_ECR_comment_repetition_desired

This message informs the *coordinator* about the *participant’s* wish to be asked again within the phase *Commenting on ECR* about comments using the message *Request_ECR_-comments*.

Parameters:

- identifier of the ECR of the *coordinator* [mandatory]

3.3.3 Use of ECR Messages within the predefined ECR Interaction Scenarios

The table below provides information on which messages are used in each of the previously described four ECR Interaction Scenario.

The following notation is used to indicate the recommended use:

- **m** message must be supported in the IS, i.e. mandatory (Note: message occurrence can vary depending on the actual IS involved)
- **cs** message use needs to be agreed upon cooperation-specifically

- **na** message use is not applicable in the IS

Some of the messages in the table are described within the ECR Reference Process only for the purpose of supporting special flow variants and exceptions. These messages are indicated in the column “special messages…”.

Table 2: Recommended use of ECR Messages in the ECR Interaction Scenarios

	Message	from participant	from coordinator	special message, which is not shown in the IS description	IS1	IS2	IS3	IS4
process oriented	*Request_initial_ECR*	x			**m**	**m**	na	**m**
	Respond_initial_ECR		x		**m**	**m**	na	**m**
	Notify_initial_ECR_rejected		x		**m**	**m**	na	**m**
	Notify_initial_ECR_accepted		x		**m**	**m**	na	**m**
	Notify_ECR_rejected		x		**m**	**m**	na	**m**
	Notify_ECR_accepted		x		**m**	**m**	na	**m**
	Notify_ECR_created		x		na	**cs**	na	**cs**
	Request_ECR_details		x		na	na	na	**m**
	Respond_ECR_details	x			na	na	na	**m**
	Notify_ECR_detailed		x	x	**cs**	**cs**	**cs**	**m**
	Request_ECR_comments		x		na	**cs**	na	**cs**
	Respond_ECR_comments	x			na	**cs**	na	**cs**
	Notify_ECR_comment_repetition_desired	x		x	na	**cs**	na	**cs**
	Request_ECR_confirmation_comments		x		na	**cs**	na	**cs**
	Respond_ECR_confirmation_comments	x			na	**cs**	na	**cs**
	Notify_ECR_commented		x	x	**cs**	**m**	**cs**	**m**
	Request_ECR_completeness_check		x	x	na	na	**cs**	na
	Respond_ECR_completeness_check	x		x	na	na	**cs**	na
	Notify_ECR_id	x			na	na	**cs**	na
	Request_ECR_acceptance		x		na	na	**m**	na
	Respond_ECR_acceptance	x			na	na	**m**	na
	Notify_ECR_decided		x		**m**	**m**	**m**	**m**
	Notify_ECR_rolled_back_to_analysis		x	x	na	**cs**	na	**cs**
	Notify_ECR_rolled_back_to_commenting		x	x	na	**cs**	na	**cs**
	Notify_ECR_canceled		x	x	**m**	**m**	**m**	**m**
	Notify_ECR_closed		x	x	**cs**	**cs**	**cs**	**m**
	Notify_ECR_update		x		na	**cs**	na	**cs**
data oriented	*Request_all_ECR_ids*	x		x	**cs**	**cs**	**cs**	**cs**
	Respond_all_ECR_ids		x	x	**cs**	**cs**	**cs**	**cs**
	Request_ECR_data	x		x	**cs**	**cs**	**cs**	**cs**
	Respond_ECR_data		x	x	**cs**	**cs**	**cs**	**cs**

3.3.4 ECR Message headers and control information

The following **additional** specifications and restrictions apply to the ECR Message header for messages of the type *request, notify and respond*, thus extending the specification given in Part 0 of this recommendation, section 6.2.

Table 3: Message headers and control information for ECR Messages

Data object	Description (types/format, sample value, meaning)
INTERACTIONID	Format*: ORGANIZATIONID "/" ID.ID "/" SUBDIALOGNR;* example: "717743322/AEA553421/1" ***Note:*** If the owner of the *ID* is the sender, the *ID.ID*_owner should identify the same organization as the *ORGANIZATIONID* (likewise for the receiver).
INTERACTIONSCENARIONAME	Exemplary values: "IS1", "IS2", "IS3", "IS4"
MESSAGENAME	Values: Name of the message, see section 3.3
KINDOFSUBDIALOG	If possible, the following values should be used to make a distinction between the different kinds of subdialogs: "GENERAL_COMMENTS" - Comments on the ECR that are not differentiated further. In this case, there should be no other subdialog concerning comments. (Note: Even with only one dialog, different kinds of comments may be exchanged in a single Request_ECR_comments or Respond_ECR_comments message (see 2.4.4). "TECHNICAL_COMMENTS" - Comments on the ECR with a technical focus, i. e. regarding the effects from the point of view of engineering (design, test, ...), manufacturing, etc. "COMMERCIAL_COMMENTS" - Comments on the ECR regarding the cost effects of the change (costs relating to parts, tools, ...) "LOGISTICS_COMMENTS" - Comments on the ECR regarding the logistical effects of the change (container changes necessary, ...) Please note that the exact definition of which objects and attributes are provided or expected is not defined. In addition, the kinds of subdialogs are not intended as a non-overlapping classification.

4 ECR Data Model

This chapter provides an introduction to the ECR data model, which reflects the specific requirements of the ECR Reference Process in the context of ECM and the ECM Data Model of Part 0. It comprises:

- A brief introduction to the description method used (4.1)
- The recommended usage of the ECM Data Model defined in Part 0 within the context of the ECR Reference Process (4.2)
- An overview of the recommended use of the data types in typical messages in the ECR application scenarios (4.3)

4.1 Description method

The ECR requirements data model is specified in an EXPRESS-G diagram accorting Part 0 section 4.4 and Appendix B.

The basis for the data model is formed by the units of information communicated within the ECR Reference Process as well as by an analysis of data models of company specific ECM application systems deployed at companies participating in the ECM working group, taking into consideration what data needs to be exchanged with partners in the context of cross-company change requests. The generalization resulted in the common ECM Data Model (see Part 0 of this recommendation).

4.2 Usage of the ECM Data Model for the ECR Reference Process

This section provides an introduction to the usage of the ECM Data Model, an introduction to the fundamental data types and the relationships between them.

No further data objects are defined as extension of the ECM Data Model as described in Part 0. Therefore the following sections describe only the usage of the general ECM Data Model object types within the context of the ECR Reference Process. Thereby only differences or extensions are described in relation to the decriptions in Part 0.

References to the phases of the ECR Reference Process help the reader to understand the data model in terms of its use in the context of the ECR Reference Process.

Each ECR must be identified uniquely by an identifier (ID.id of HEADER.id) and the organization the identifier is defined by (ID.id_owner).

The data types ID and HEADER represent the master data (identifier, description of the change and its context, responsibilities and processing status) of a proposed change request, which is constantly refined during the course of the ECR process by an increasing level of detail in the descriptions. The ID.process_phase is defined as 'ECR'. An initial ECR is identified by HEADER.status.status='pending' and is used only in the phase *INQUIRY_OF_ECR* of the ECR Reference Process to indicate that the ECR is identified by a *participant* but still not accepted by the *coordinator*. Only if the ECR is accepted by the *coordinator,* will the attributes of the initial ECR be copied to an HEADER which has an identifier of the *coordinator* and a status shift to 'initiated'. Both ECRS can be related to each other by an ID_RELATIONSHIP to define the connection between the requested and accepted ECRS.

A change request can be proposed either within the company itself (ID and HEADER) or by partners. If a change is proposed by a partner adopting the role of *participant*, the proposal is described using initial HEADER. In a case such as this, the *coordinator* decides whether to create an own change request (new ID and HEADER) corresponding to initial ID and HEADER of the partner or to reject it. In other words, the initial HEADER indicates an external request by a *participant* and the HEADER which is identified by the *coordinator* as .id_owner indicates the request by the *coordinator*. The data type ID_RELATIONSHIP relates the identifiers of both change requests.

An initial HEADER must contain the following basic information on an ECR:

- identifier of the ECR (ID.id) and
- at least a description of the intended change (HEADER.description).

Further detailed information on an ECR is permitted for an initial ECR HEADER but is not required.

The data type HEADER of .status 'pending' is the key object of the *Inquiry of ECR* phase, and the data type HEADER of .status 'initiated' is the start object of the *Creation of ECR* phase in the ECR Reference Process.

There is no explicit version mechanism for change requests.

The target of an ECR (HEADER.description) can be described using a 'flat' text or a structured data type (CHANGE_DESCRIPTION).

***Note*:** *The data models underlying most of the ECM application systems investigated do not include any explicit representation of* participant *proposals for changes in the form described above. This means that typically phase 1 of the ECR Reference Process is currently not administered in the ECM application system. This distinction is, however, necessary from the point of view of the process.*

The following properties of the HEADER information are applied specifically for an ECR:

- **id:** Unique identification of an ECR by means of an identifier (ID.id) and the scope of validity of the identifier (ID.id_owner)
- **name:** Name of the ECR as a short description of the problem
- **change_process_type**: Classification of the ECR (more than one criteria allowed)
- **leading_change_scope:** Describes the main scope of the change requested in the ECR. In general, it implicitly defines responsibility for the ECR
- **coordinator_contact:** List of contacts (persons and/or organizations) on the side of the *coordinator*. These are communicated along with the ECR as the relevant points of contact for defined topics. Their responsibilities derive from the role they play in the context of the ECR. If only one contact is specified, this contact is the only one responsible for any internal coordination required on the side of the *coordinator*.

Note: *If EC_manager is not used by the coordinator, the participant must still be provided with at least one contact in coordinator_contact.*

- **participant_contact:** List of contacts (persons and/or organizations) on the side of the *participant*. These are communicated along with the ECR as the relevant points of contact for defined topics. Their responsibilities derive from the role they play in the context of the ECR. If only one contact is specified, this contact is the only one responsible for any internal coordination required on the side of the *participant*.

  Note: *If the* participant *requests the ECR and does not specify the requestor, the* participant *must provide the* coordinator *with at least one contact in participant_contact. The same is required if the* participant *is not the* Requestor*.*

- **status:** Describes the current processing status of the ECR in the context of the ECR Reference Process

  Use one of the following values for STATUS.status:

  - **'pending':** The initial ECR in question has been submitted (Phase 1: *Inquiry of ECR*)
  - **'initiated':** *Inquiry of ECR* has been completed i.e. the initial ECR was approved for Phase 2; an ECR is now created a decision is again made as to whether to continue with the ECR Process (Phase 2: *Creation of ECR*)
  - **'created':** *Creation of ECR* has been completed and it was decided that the ECR was to be processed further; the ECR is now being analyzed from a technical point of view (Phase 3: *Technical Analysis of ECR*)
  - **'detailed':** *Technical Analysis of ECR* has been completed; the ECR is now being commented on (Phase 4: *Commenting on ECR*)
  - **'commented':** *Commenting on the ECR* has been completed; approval of the ECR is now taking place (Phase 5: *Approval of ECR*)
  - **'approved':** The ECR has been approved; this is a final stage in the ECR process.
  - **'disapproved':** The ECR was rejected in Phase 1 (as an initial ECR), in Phase 2, or in Phase 5 (note: the originating phase is described using acceptance); this is a final stage in the ECR process.
  - **'cancelled':** The ECR was cancelled between Phase 2 to Phase 5 (note: the originating phase is described using acceptance); this is a final state in the ECR process
  - **'closed':** The ECR is closed, if the ECR is closed at both sides (*coordinator* and *participant*).

Detailed technical and commercial information relating to an ECR is represented by DETAIL. This data type can only be used (instantiated) as one of its specializations.

The data type DETAIL is the key object of the *Technical Analysis of ECR* phase in the ECR Reference Process. In the context of the ECR Reference Process, the definition of DETAIL is typically exchanged with the partners affected (*Analysis Performers*) during the phase. Once the phase *Technical Analysis of ECR* has been concluded, no more changes are permitted. (DETAIL may also be used before the phase *Technical Analysis of ECR* to exchange preliminary detail data, e.g. details given in a *Request_Initial_ECR* by a *Requestor.*) The distinction between general details (GENERAL_DETAIL) and specific details (SPECIFIC_DETAIL) is used to distinguish between information which is valid for the entire ECR (general) and information which is only valid for a specific section of it. Specific details have a higher priority and may be used to specify groups of affected objects or to define a replaced-by relationship in detail.

For example, the delivery date for a prototype of an individual part can be set earlier than the delivery date for the prototype containing all the changes covered by the ECR. In a case such as this, the delivery date for the prototype is specified as GENERAL_DETAIL and the delivery date for the individual part is specified with a higher priority as SPECIFIC_DETAIL.

The follwing properties of DETAIL information are applied specifically for an ECR:

- **affected objects:** Objects that are affected by the ECR. A distinction is made between input objects (before the change) and output objects (after the change).
- **status**: Describes the state of detailing of the ECR
- **kind**: Classifies an ECR
- **tasks:** Deadlines which are of significance before the decision is taken or during processing of the proposed change.
- **attachments**: Data files assigned to an ECR as general additional information

Rule: *For each ECR ID, there is a maximum of one instance of GENERAL_DETAIL and no instance, one instance or several instances of SPECIFIC_DETAIL. If instances of SPECIFIC_DETAIL exist, there must also be (exactly) one instance of GENERAL_DETAIL.*

The data type COMMENT represents comments on an ECR in general or comments on details. It includes statements regarding the affected objects, calculation, changes in weight and/or statements concerning feasibility and consequences of the change. The sequence of comments is defined using COMMENT_RELATIONSHIP.

The data type COMMENT is the key object of the *Commenting on ECR* phase in the ECR Reference Process. In the context of the ECR Reference Process, the definition of COMMENT is typically exchanged with affected partners (*Comment Performers*) during the phase. Once the *Commenting of ECR* phase has been completed, no more changes are permitted. (COMMENT may also be used before the phase *Commenting of ECR* in order to exchange preliminary comment data, e. g. comments provided in a *Request_Initial_ECR* message by a Requestor.)

The following properties of COMMENT information are applied specifically for an ECR:

- **affected objects**: The comment is given for the identified objects only
- **status:** Status of the given comment. There are two possible status: ‘preliminary’ and ‘final’ for supporting an iterative commenting process.
- **kind**: Classifies the comments as technical or review-evaluation comments with CLASS.id = 'ECR comment classification' and PROPERTY.property_type = 'ECR kind of comment'
- **tasks:** Deadlines which are of significance before the decision is taken or during processing of the proposed change.
- **attachments**: Data files assigned to an ECR as general additional information

ACCEPTANCE represents the result of a decision about an ECR by the *coordinator* or an involved participant (as an External Approval Performer in IS3) related to a defined ECR process status.

The data type ACCEPTANCE (in particular the tupel of the attributes .decision and .related_status) defines the decision status of an ECR within the ECR Reference Process (see Figure 33). As shown in that state chart, ACCEPTANCE may be set by the *coordinator* to the related_status.status ‘pending’, ‘initiated’ or ‘commented’.

From the *coordinator’s* point of view, an ECR can be accepted or rejected. In the ‘commented’ phase, it can also be decided that it requires a revision, starting back from Technical Analysis of ECR, Commenting on ECR or Approval of ECR. From the point of view of a *participant* as an External Approval Performer in IS3, the ECR can be accepted, rejected, or incomplete.

A sequence of acceptances is represented by ACCEPTANCE_RELATIONSHIP.

The following properties of ACCEPTANCE information are applied specifically for an ECR:

- **decision:** permitted values are 'accepted', 'rejected', ' 'revision needed back to analysis', 'revision needed back to approval', or 'revision needed back to commenting'
- **related status:** related status of the ECR within the ECR Reference Process. In the context of ACCEPTANCE, the values permitted for Status.status are 'pending', 'initiated', or 'commented'.

The common ECM data model defines a data type STATUS. In order to use it in the context of the specific ECR data model, it must be completed by the ECR Reference Process (see Figure 37).

Note**:** *As indicated in the diagram, if the ECR is canceled or rolled-back, the relevant message (Notify_ECR_canceled, Notify_ECR_rolled_back_to_analysis, …) makes reference to a Status object which provides a value for Status.status which indicates the originating state. For example, if an ECR is aborted with the status “detailed”, this should be the value of Status.status for the most recent Status object in the ECR.*

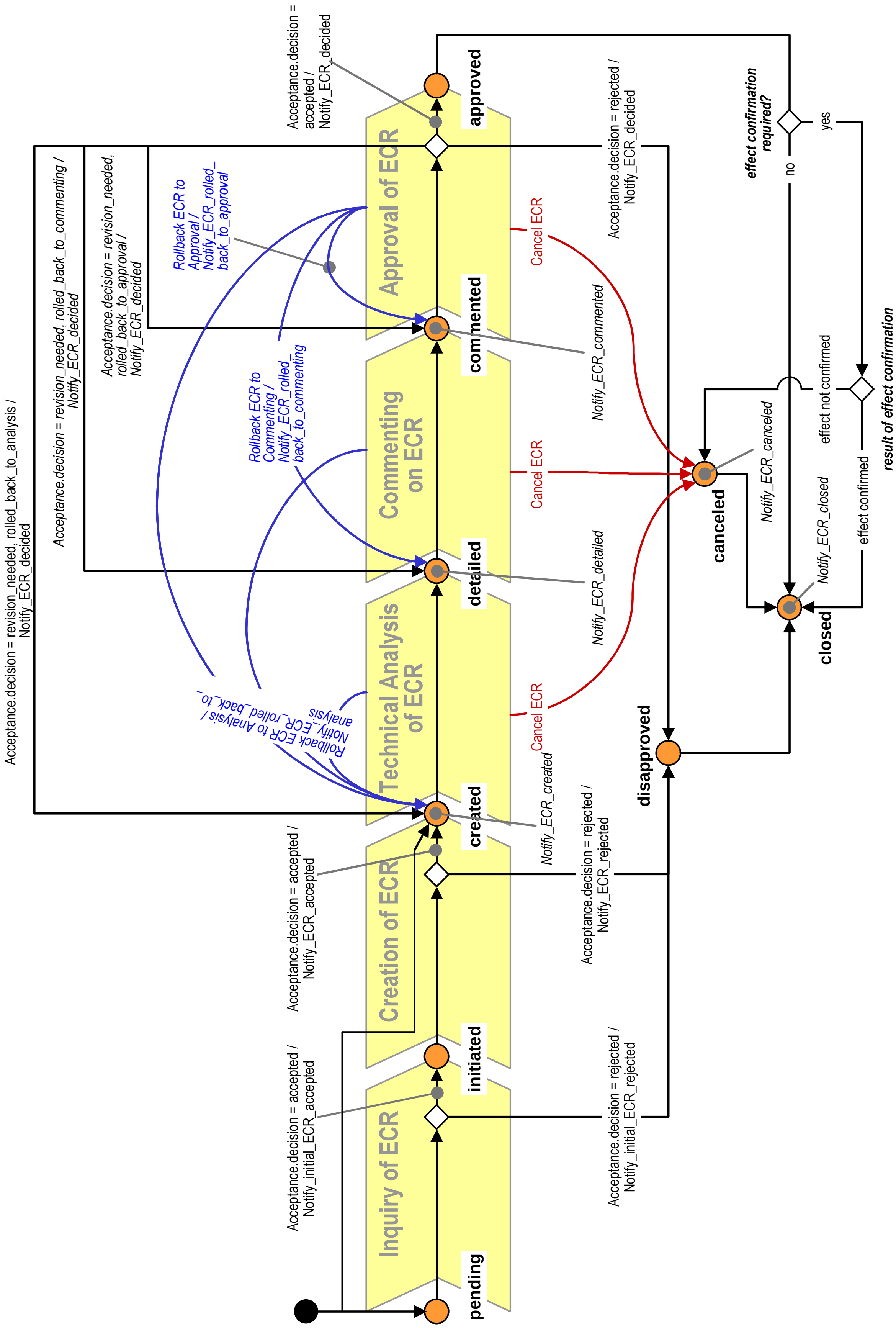

Figure 33: ECR states and state transitions

4.3 Usage of ECM Data Model objects within ECR Messages

The table below provides information on which data objects are typically used as content for each of the ECR Messages. The following notation is used to indicate the recommended use:

Table 4: Classification for the recommended use of data objects

m	at least one instance of the data object is required, i.e. is mandatory, within the message (including its mandatory attributes)
o	optional information, which may be instantiated within the message depending on the use case and content of the ECR
NE	data object is not expected in the message, e.g. it may be defined as "mandatory" or "optional" if requested on a cooperation-specific basis

Table 5: Recommended use of data objects as content/payload for ECR Messages

	Message	from participant	from coordinator	special message, which is not shown in the IS description	ID of participant	Initial Header of participant	ID_relationship	Initial Header of coordinator	ID of coordinator	Header	Header.status	General_detail	Specific_detail	Comment	Acceptance
process oriented	*Request_initial_ECR*	x			m	m	o	ne	ne	ne	o	o	o	o	ne
	Respond_initial_ECR		x		m	ne	m	m	m	ne	o	o	o	o	ne
	Notify_initial_ECR_rejected		x		ne	ne	ne	m	m	ne	m	ne	ne	ne	m
	Notify_initial_ECR_accepted		x		ne	ne	o	m	m	ne	m	o	o	o	m
	Notify_ECR_rejected		x		ne	ne	o	ne	m	m	m	o	o	o	m
	Notify_ECR_accepted		x		ne	ne	o	ne	m	m	m	o	o	o	m
	Notify_ECR_created		x		ne	ne	o	ne	m	m	m	o	o	o	ne
	Request_ECR_details		x		ne	ne	o	ne	m	m	m	o	o	o	o
	Respond_ECR_details	x			ne	ne	o	ne	m	o	ne	m	o	o	ne
	Notify_ECR_detailed		x	x	ne	ne	o	ne	m	o	m	o	o	ne	ne
	Request_ECR_comments		x		ne	ne	o	ne	m	m	m	m	o	o	o
	Respond_ECR_comments	x			ne	ne	o	ne	m	o	ne	o	o	m	ne
	Notify_ECR_comment_repetition_desired	x		x	ne	ne	ne	ne	m	o	o	ne	ne	ne	ne
	Request_ECR_confirmation_comments		x		ne	ne	o	ne	m	m	m	m	o	m	ne
	Respond_ECR_confirmation_comments	x			ne	ne	o	ne	m	o	ne	o	o	m	ne
	Notify_ECR_commented		x	x	ne	ne	o	ne	m	o	m	o	o	o	ne
	Request_ECR_completeness_check		x	x	ne	ne	o	ne	m	m	m	m	o	m	ne
	Respond_ECR_completeness_check	x		x	ne	ne	o	ne	m	o	ne	o	o	o	m
	Notify_ECR_id	x			m	ne	m	ne	m	o	ne	ne	ne	ne	ne
	Request_ECR_acceptance		x		ne	ne	o	ne	m	m	m	m	o	m	ne
	Respond_ECR_acceptance	x			ne	ne	o	ne	m	o	ne	o	o	o	m
	Notify_ECR_decided		x		ne	ne	o	ne	m	m	m	o	o	o	m
	Notify_ECR_rolled_back_to_analysis		x	x	ne	ne	ne	ne	m	o	m	ne	ne	ne	m
	Notify_ECR_rolled_back_to_commenting		x	x	ne	ne	ne	ne	m	o	m	ne	ne	ne	m
	Notify_ECR_canceled		x	x	ne	ne	o	ne	m	o	m	o	o	o	m
	Notify_ECR_closed		x	x	ne	ne	o	ne	m	o	m	o	o	o	o
	Notify_ECR_update		x		ne	ne	o	ne	m	m	m	o	o	o	o
data oriented	*Request_all_ECR_ids*	x		x	ne	ne	ne	ne	ne	ne	o	ne	ne	ne	o
	Respond_all_ECR_ids		x	x	o	ne	o	ne	m	m	o	ne	ne	ne	o
	Request_ECR_data	x		x	ne	ne	ne	ne	m	ne	ne	ne	ne	ne	ne
	Respond_ECR_data		x	x	o	ne	o	ne	m	m	o	o	o	o	o

Appendix A: ECR Interaction Scenario Protocol Specifications

A.1 Purpose & Overview

This document is intended to provide a description of the protocol specification for ECR Interaction Scenarios, i.e. describes the message sequences that are permitted between the *coordinator* and *participant*. A more precise definition, from a technical point of view, on the message sequences of the ECR Reference Process is provided in Section 2.4 of this document. The ECR Interaction Scenarios are described more detailed in Section 3.2.

This protocol specification makes use of those parts of the ECR Reference Process that are relevant to and need to be supported by the *coordinator* and *participant* if they want to implement or use a specific ECR Interaction Scenario, i.e. message send/receive actions and control flow. The ECR Interaction Scenario protocol specification has been extended to include additional messages defined in the ECR Reference Process, e.g. for rollbacks, the use of revision, permitted types of comment, etc. Some additional restrictions are given that concern the mandatory and cooperation-specific protocol requirements defined in Section 2.2.4 (use of estimation-based approval).

In the following, a public view[1] is given for every of the aforementioned ECR Interaction Scenarios using UML activity charts (see Appendix A of Part 0) for the *coordinator* as well as for the *participant*. Although the public views for the *coordinator* and participant are for the most parts symmetrical, a separate specification is provided for each for reasons of accuracy. Where necessary due to space constraints or complexity, some special cases of the protocols (i.e. special details) are depicted in separate diagrams (i.e. the contents described in the diagrams for an ECR Iinteraction Scenario have to be implemented in order to satisfy the ECR Interaction Scenario Protocol Specification).

With regard to the ECR Interaction Scenario Protocols Specification, this recommendation leaves a certain amount of room for this to be defined in more detail between *coordinator* and *participant* or by referring to other recommendations. The annotation “<<cooperation specific input>>” is used in the decision nodes in the diagrams to indicate where cooperation-specific input is needed. This also allows a distinction to be made between cooperation-specific input and decision input (indicated by “<<decision input>>”). Decision input is specified during a specific instance of an ECR Interaction Scenario e.g. by whether or not the *participant* has requested the initial ECR or how the acceptance was decided.

The next section provides an overview of the cooperation-specific input parameters that are used for cooperation-specific input decisions. It can serve as a basis for checking whether all the required specifications regarding message sequence have been agreed upon by the *coordinator* and *participant*.

[1] A public view represents the flow of ECM Messages along an ECM Interaction Scenario.

A.2 Cooperation-specific parameters

In the protocol specifications for the ECR Interaction Scenarios, some decisions are based on the parameters for cooperation-specific input (see Figure C-1). In the table below, the co-operation-specific parameters are listed in the first column. The other columns indicate the relevance of the parameters with regard to a specific interaction scenario, the messages that are mainly affected by the parameters, possible values, and an explanation of the parameters.

Please note the following:

- The meaning of an entry is defined here only within the context of the Interaction Scenario Protocol Specification. The definition has no impact on the process or implications for the data model.

Table 6: Cooperation-specific input parameters

Cooperation specific input	IS1	IS2	IS3	IS4	Mainly affected messages	Possible values	Explanation
Kind of IS2 Support	na	cs	na	na	*(all IS2 messages)*	SET of {"coord-triggerable", "part-triggerable"}	determines whether in IS2 the coordinator may trigger the interaction or (alternatively/ as well) the participant may trigger.
Kind of IS4 Support	na	na	na	cs	*(all IS4 messages)*	SET of {"coord-triggerable", "part-triggerable"}	determines whether in IS4 the coordinator may trigger the interaction or (alternatively/ as well) the participant may trigger.
With Notification of ECR creation	na	cs	na	cs	Notify_ECR_creation	TRUE/FALSE	determines whether the coordinator sends Notify_ECR_creation or not
With ECR Updates	na	cs	na	cs	Notify_ECR_update	TRUE/FALSE	determines whether the coordinator may send Notify_ECR_update or not
With Rollbacks	na	cs	na	cs	Notify_ECR_rolled_back _to_analysis Notify_ECR_rolled_back _to_commenting	TRUE/FALSE	determines whether the coordinator may send Notify_ECR_rolled_back_to_analysis and …_to_commenting or not
With incomplete details	na	na	na	cs	Request_ECR_details Respond_ECR_details	TRUE/FALSE	determines whether the coordinator may send Request_ECR_details with incomplete details and the participant may answer those with Respond_ECR_details with incomplete details as well (cf. 2.4.3); or not
Kinds of Subdialogs	na	cs	na	cs	Request_ECR_comments Respond_ECR_comments Notify_ECR_ comment_repetition_desired	SET of ({"GENERAL_COMMENTS", "TECHNICAL_COMMENTS", "COMMERCIAL_COMMENTS", …}	determines which kinds of subdialogs can be used for which comment messages. Note that Kinds of Subdialogs may also be an empty set, i. e. no comments are exchanged.
With estimation-based approval	na	cs	na	cs	Request_ECR_comments Respond_ECR_comments	TRUE/FALSE	determines whether the coordinator may use the ECR process variant "estimation based approval" to omit the requesting of commercial comments by the participant
With coordinator requesting comments	na	na	na	cs	Request_ECR_comments Respond_ECR_comments	TRUE/FALSE	determines wheter the coordinator requests comments or not
With Notification of ECR comment repetition	na	cs	na	cs	Notify_ECR_comment _repetition_desired	TRUE/FALSE	determines wheter the participant may send Notify_ECR_comment_repetition_desired or not
With coordinator requesting confirmation	na	cs	na	cs	Request_ECR _confirmation_comments Respond_ECR _confirmation_comments	TRUE/FALSE	determines wheter the coordinator requests conformation comments or not
With completeness check	na	na	cs	na	Request_ECR _completeness_check Respond_ECR _completeness_check	TRUE/FALSE	determines whether the participant requests a completeness check during Approval of ECR or not.
With switch to participant ECR id	na	na	cs	na	Notify_ECR_id	TRUE/FALSE	determines whether the participant sends Notify_ECR_id to and this is used as MessageHeader.InteractionID from then on.

cs - cooperation specific - message use needs to be agreed upon cooperation-specifically
na - not applicable - definition of cooperation specific input parameter is irrelevant in this IS.

A.3 Protocol Specifications

Note: *The ECR Messages Notify_ECR_detailed, Notify_ECR_commented, Notify_ECR_-closed, and Notify_response_deferred are not explicitly illustrated in the following ECR Interaction Scenario Protocol Specification diagrams.*

A.3.1 ECR Interaction Scenario 1 Protocol Specification

In the following two diagrams, the public view for *coordinator* and *participant* for *IS1* is specified.

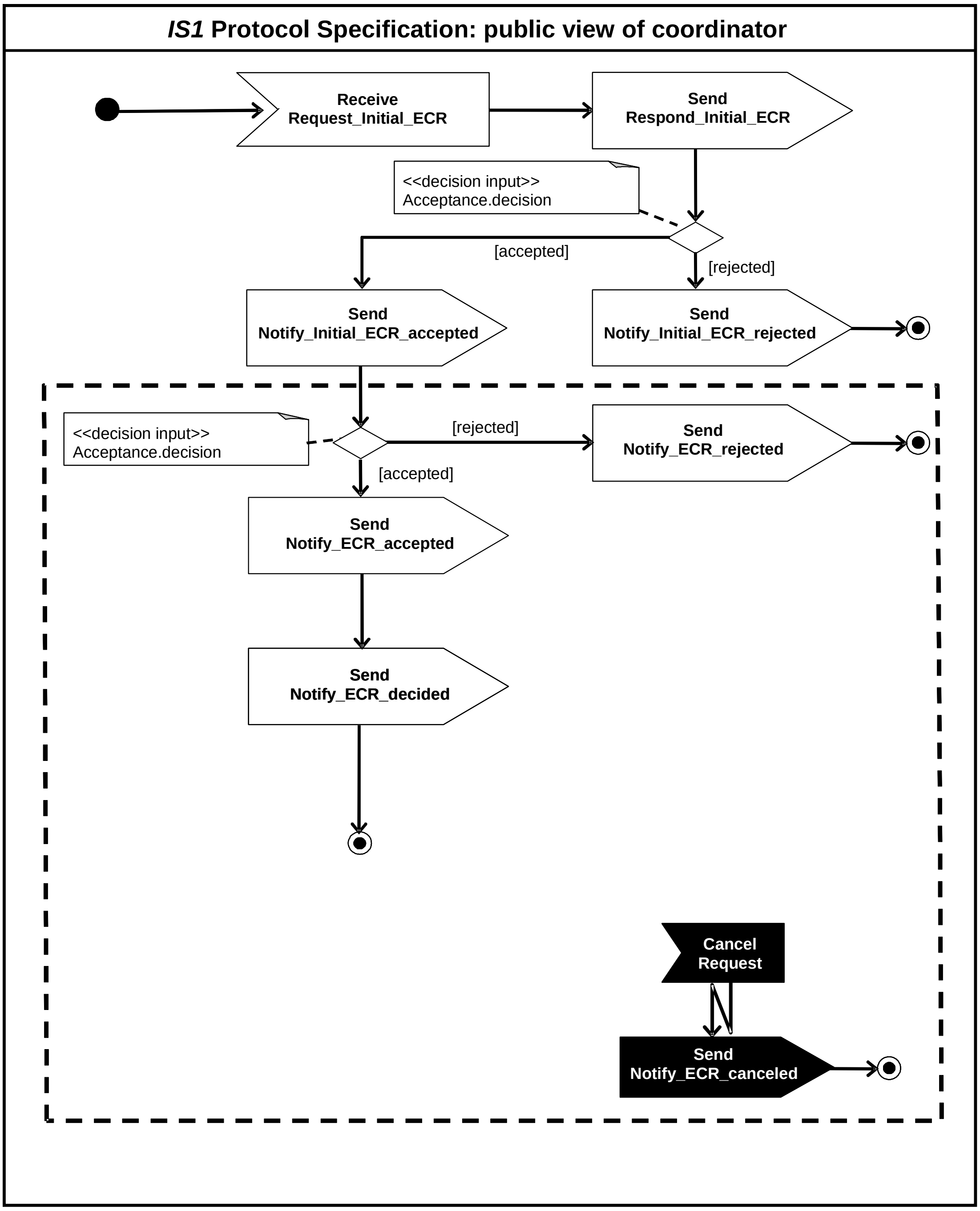

Figure 34: IS1 Protocol Specification: public view of coordinator

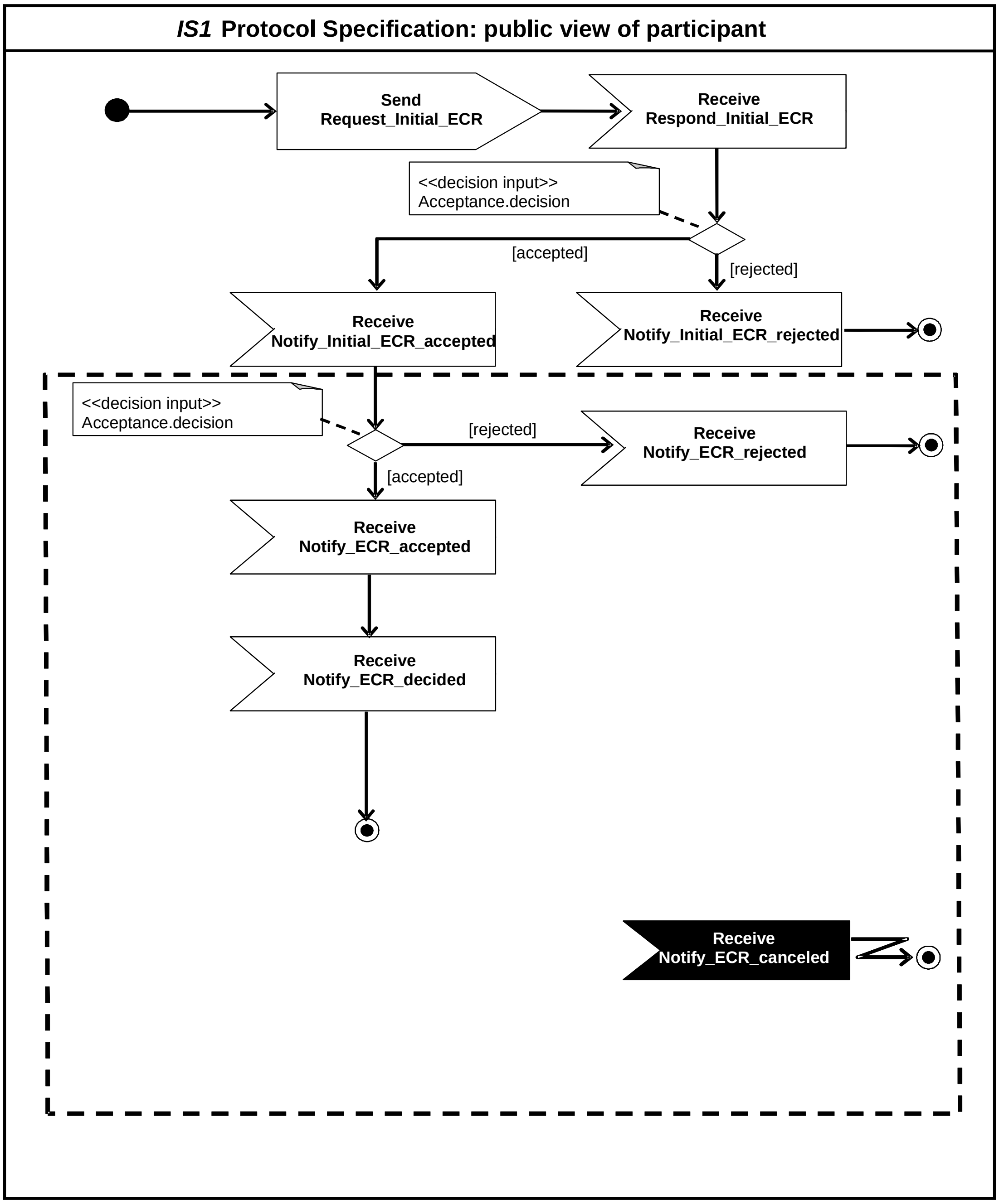

Figure 35: IS1 Protocol Specification: public view of participant

A.3.2 ECR Interaction Scenario 2 Protocol Specification

In the following two diagrams, the public view for *coordinator* and *participant* for *IS2* is first of all specified without any special cases. The subsequent two diagrams contain a specification for special cases.

Note: *The first message for a participant not requesting the ECR may be* Notify_ECR_-rolled_back_to_analysis*,* Notify_ECR_rolled_back_to_commen-ting *or* Notify_ECR_-canceled*. This may occur if there is no early notification of the ECR (Notification of*

ECR creation = FALSE) and cancel or rollback takes place before Request_ECR_-comments *is sent.*

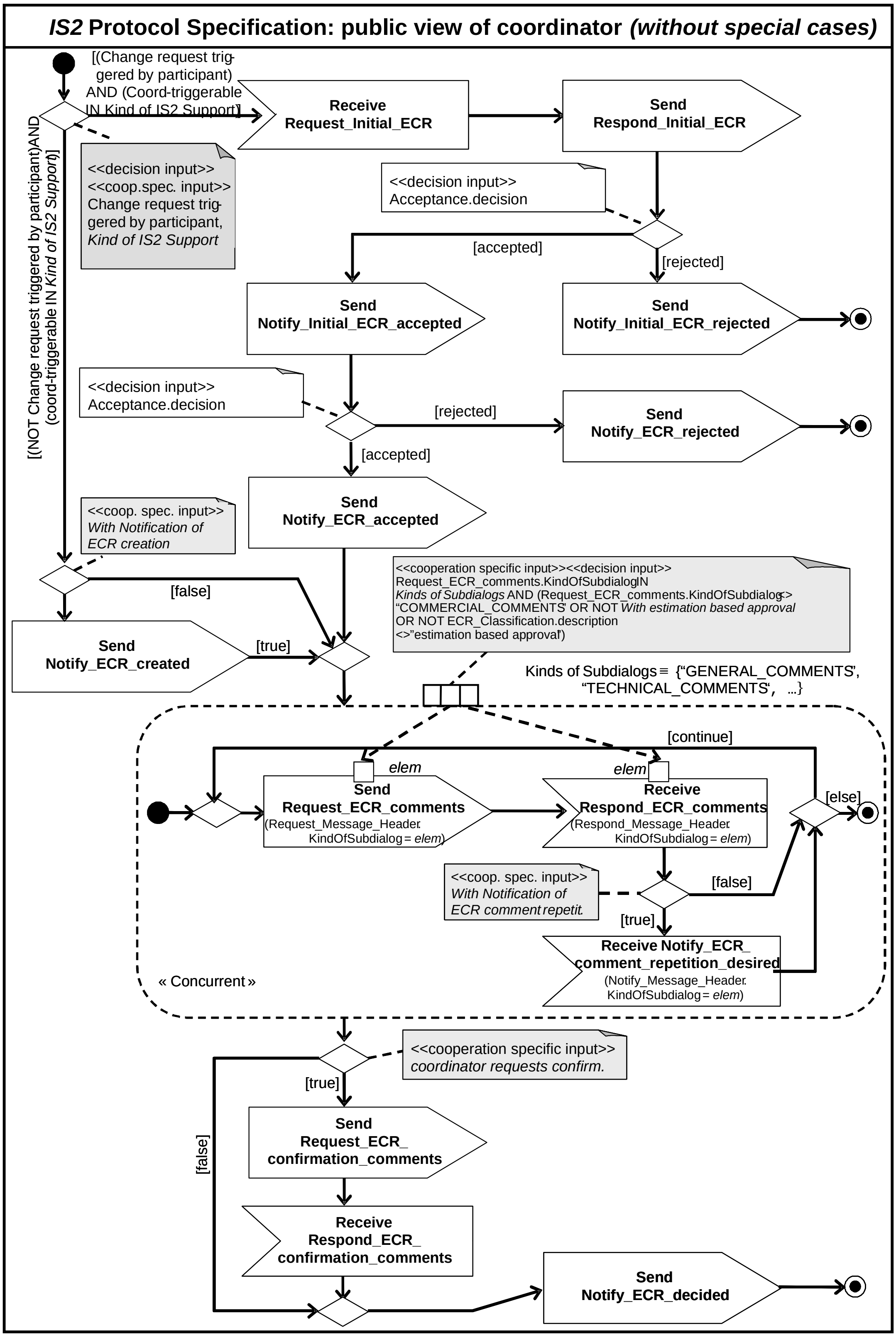

Figure 36: IS2 Protocol Specification: public view of coordinator (without special cases)

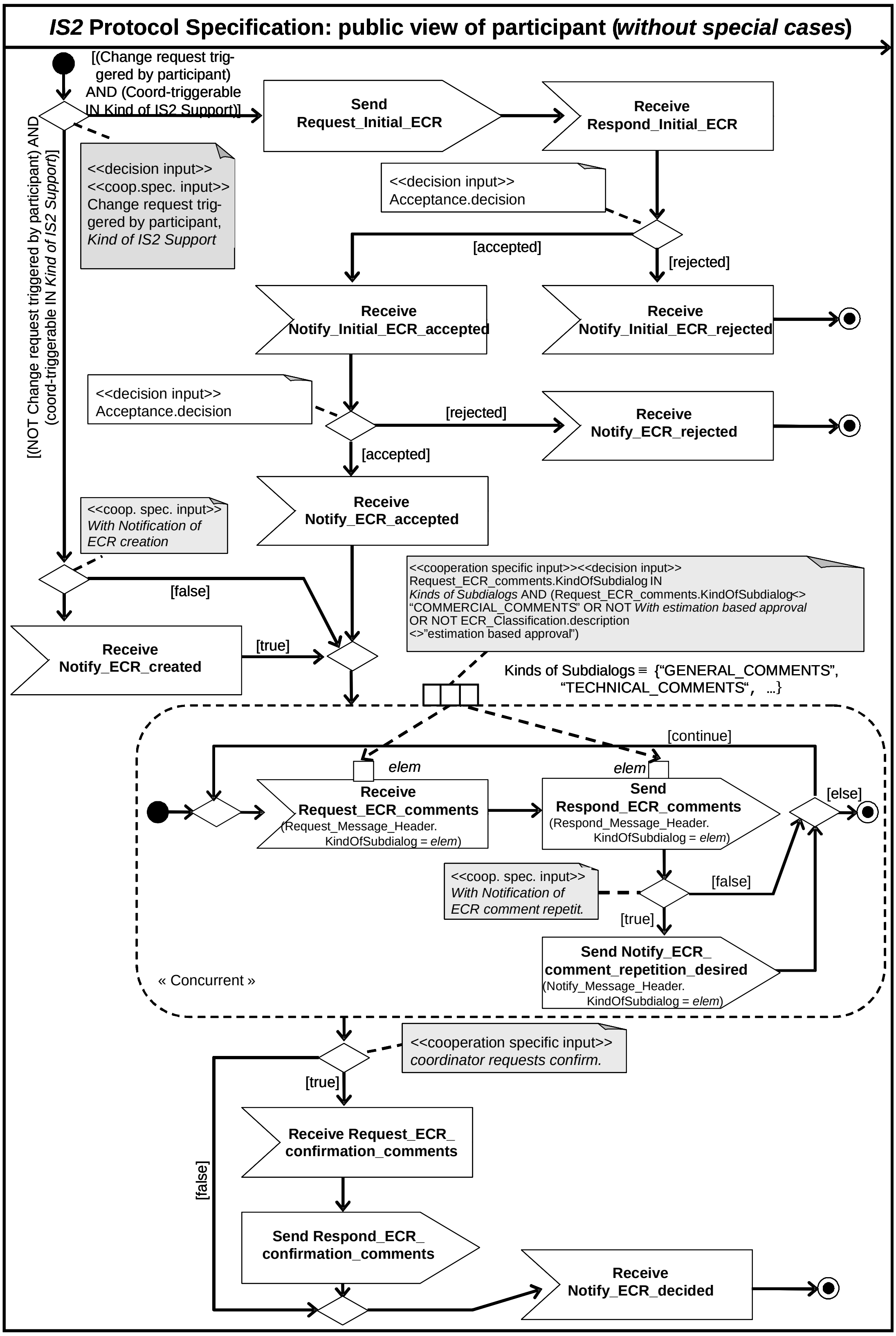

Figure 37: IS2 Protocol Specification: public view of participant (without special cases)

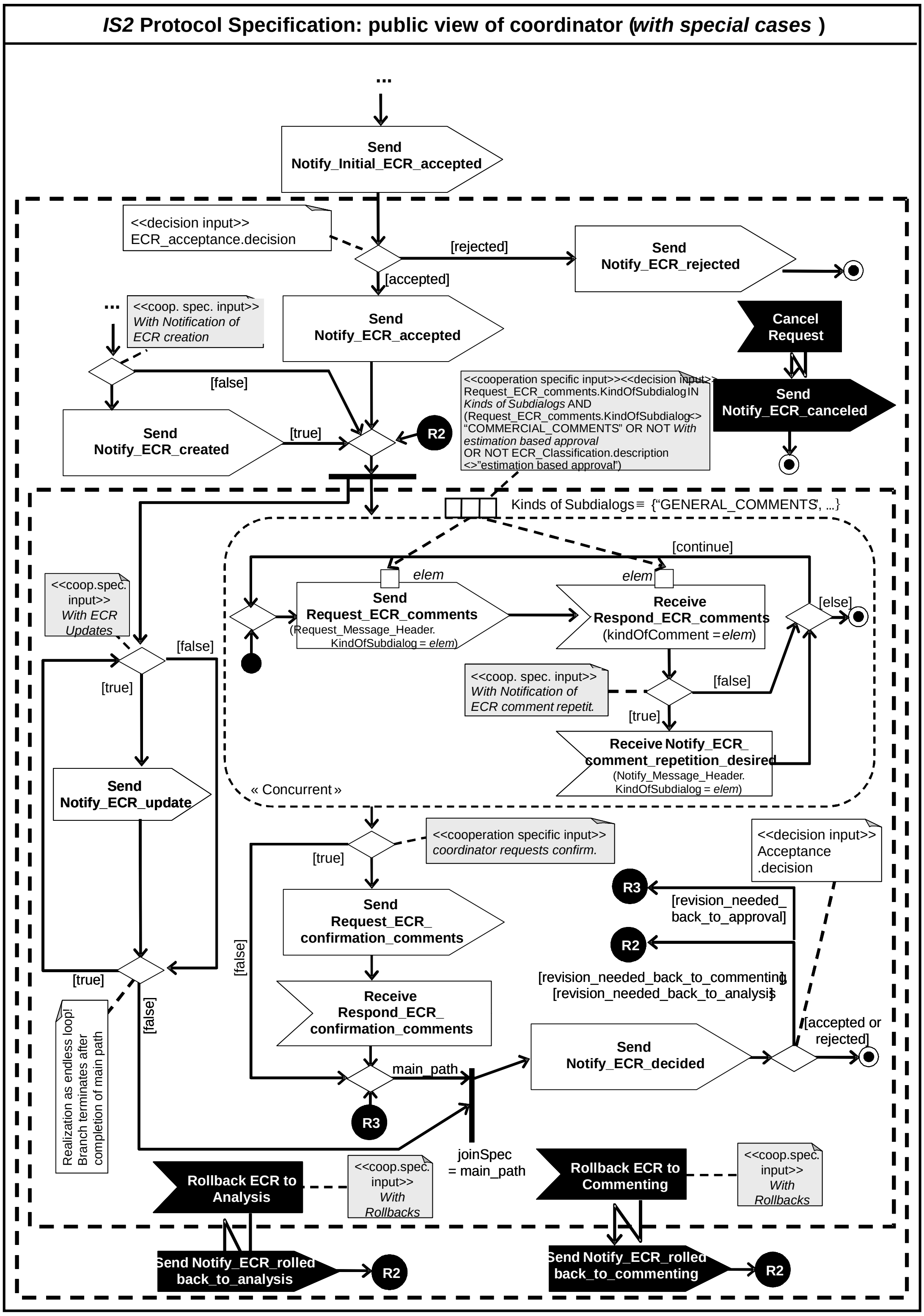

Figure 38: IS2 Protocol Specification: public view of coordinator (with special cases)

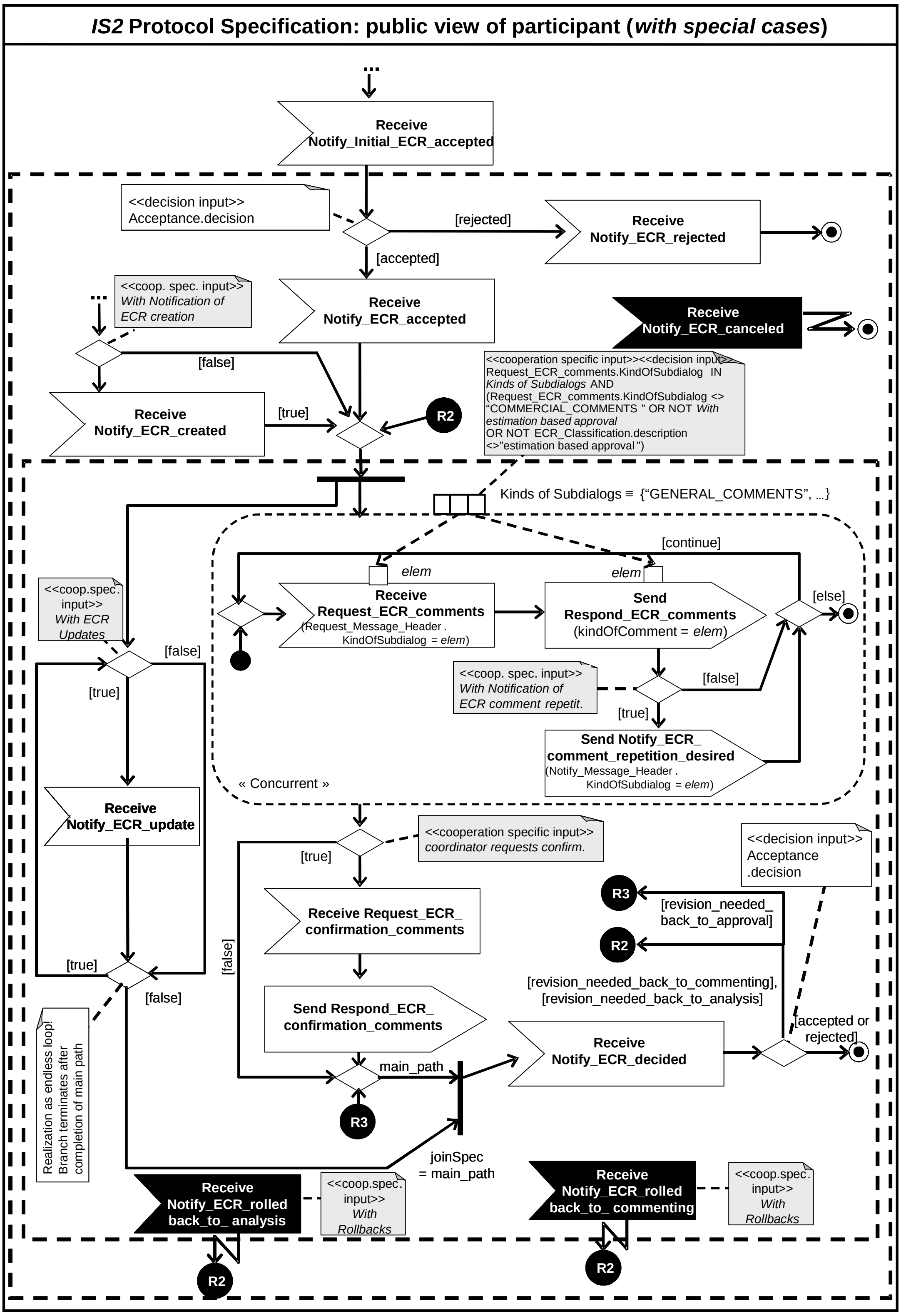

Figure 39: IS2 Protocol Specification: public view of participant (with special cases)

A.3.3 ECR Interaction Scenario 3 Protocol Specification

In the following two diagrams, the public view for *coordinator* and *participant* for *IS3* is specified.

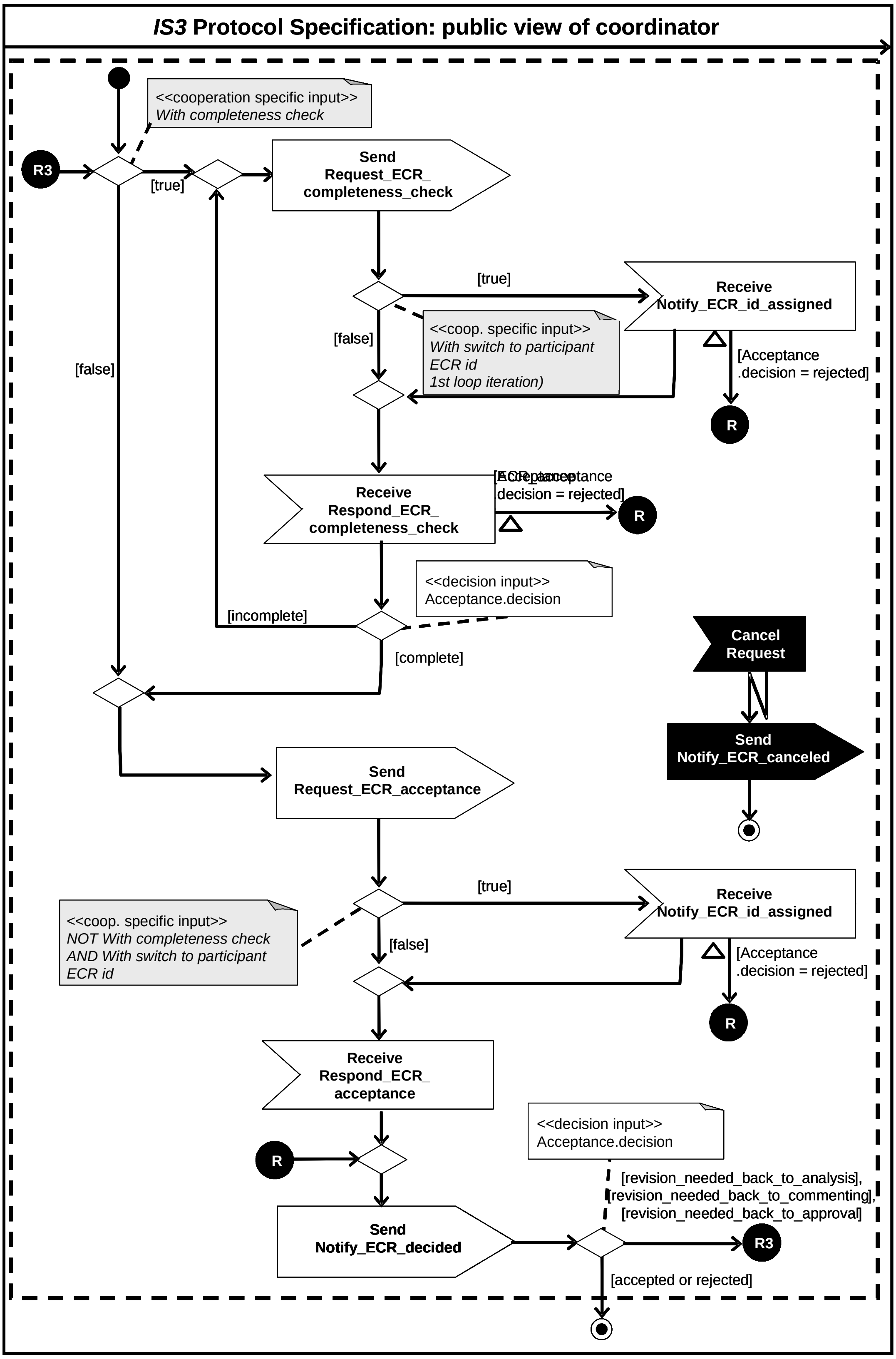

Figure 40: IS3 Protocol Specification: public view of coordinator

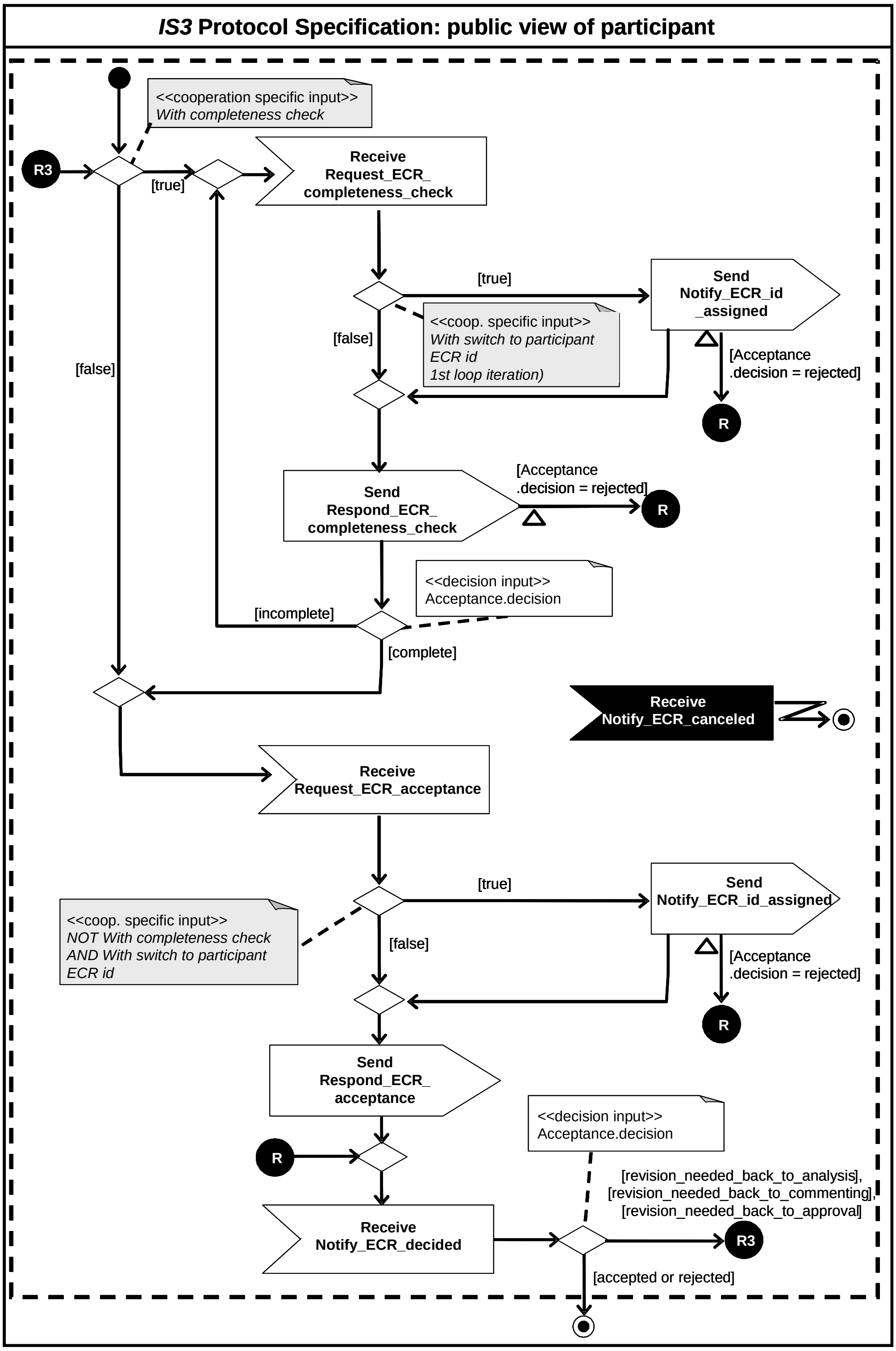

Figure 41: IS3 Protocol Specification: public view of participant

A.3.4 ECR Interaction Scenario 4 Protocol Specification

In the following two diagrams, the public view for *coordinator* and *participant* for *IS4* is first of all specified first without any special cases. The subsequent two diagrams contain a specification for special cases.

Note: *The first message for a* participant *not requesting the ECR may be* Notify_ECR_rolled_back_to_analysis *or* Notify_ECR_canceled*. This may occur if there is no early notification of the ECR (Notification of ECR creation = FALSE) and cancel or rollback takes place before* Request_ECR_details *is sent.*

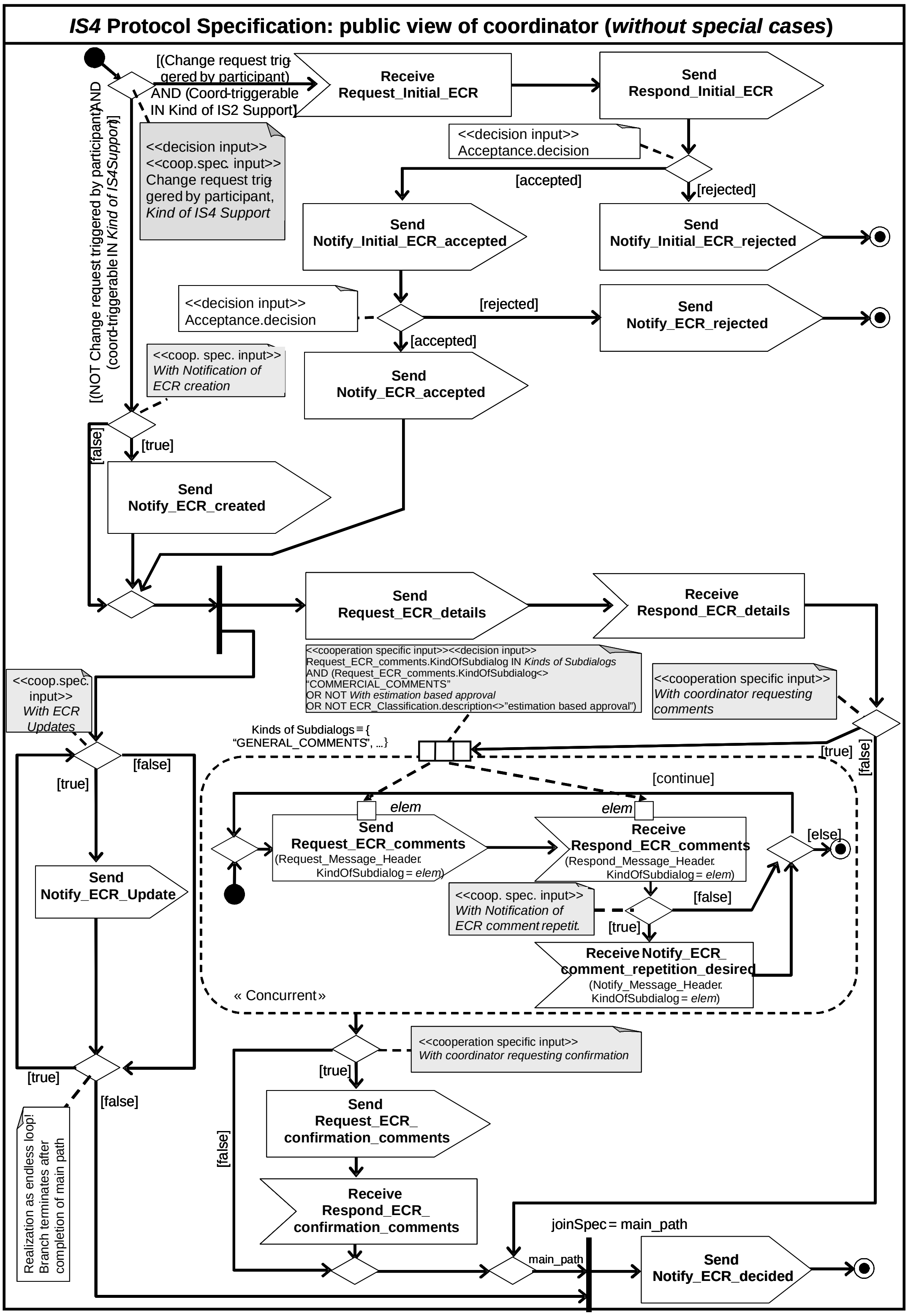

Figure 42: IS4 Protocol Specification: public view of coordinator (without special cases)

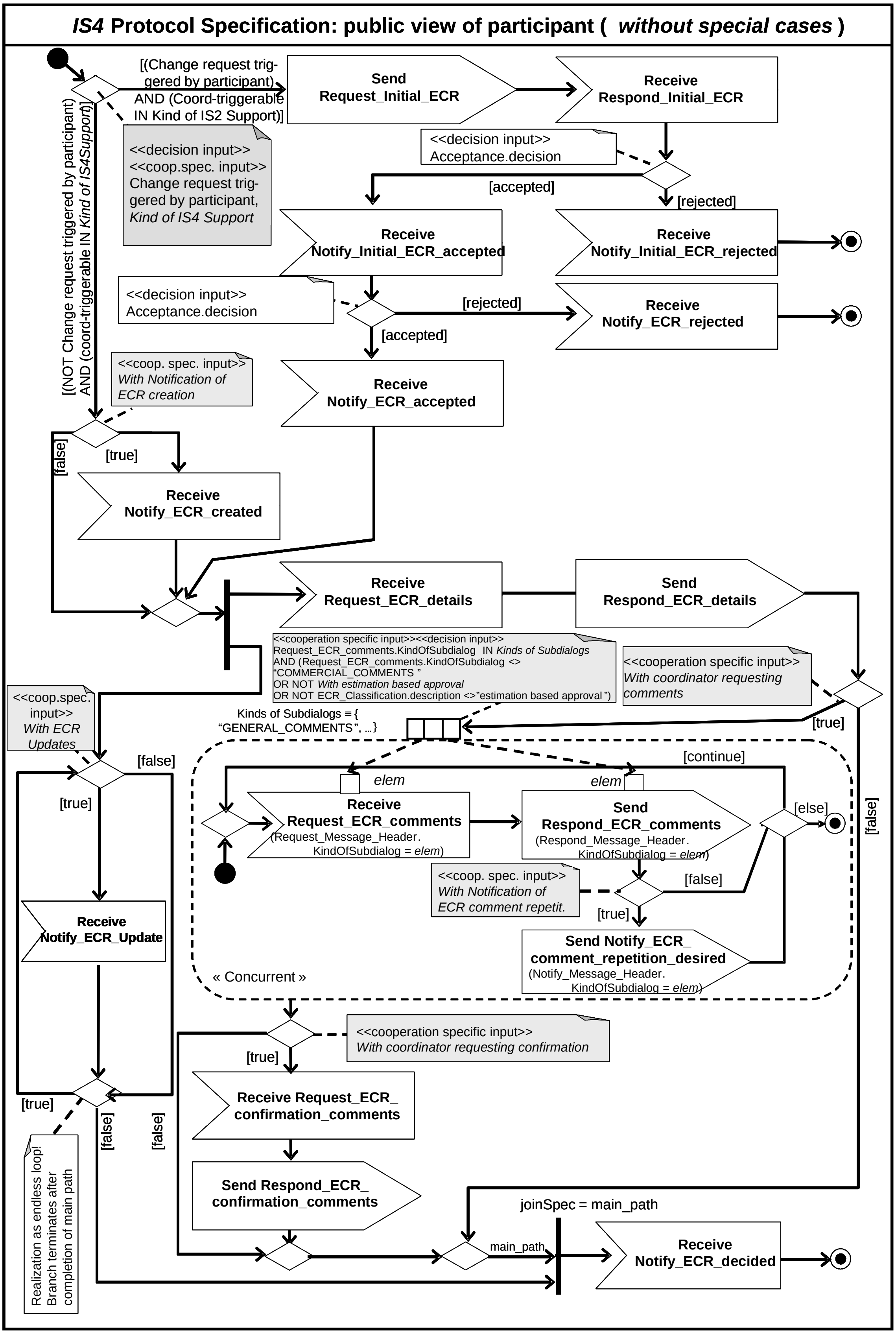

Figure 43: IS4 Protocol Specification: public view of participant (without special cases)

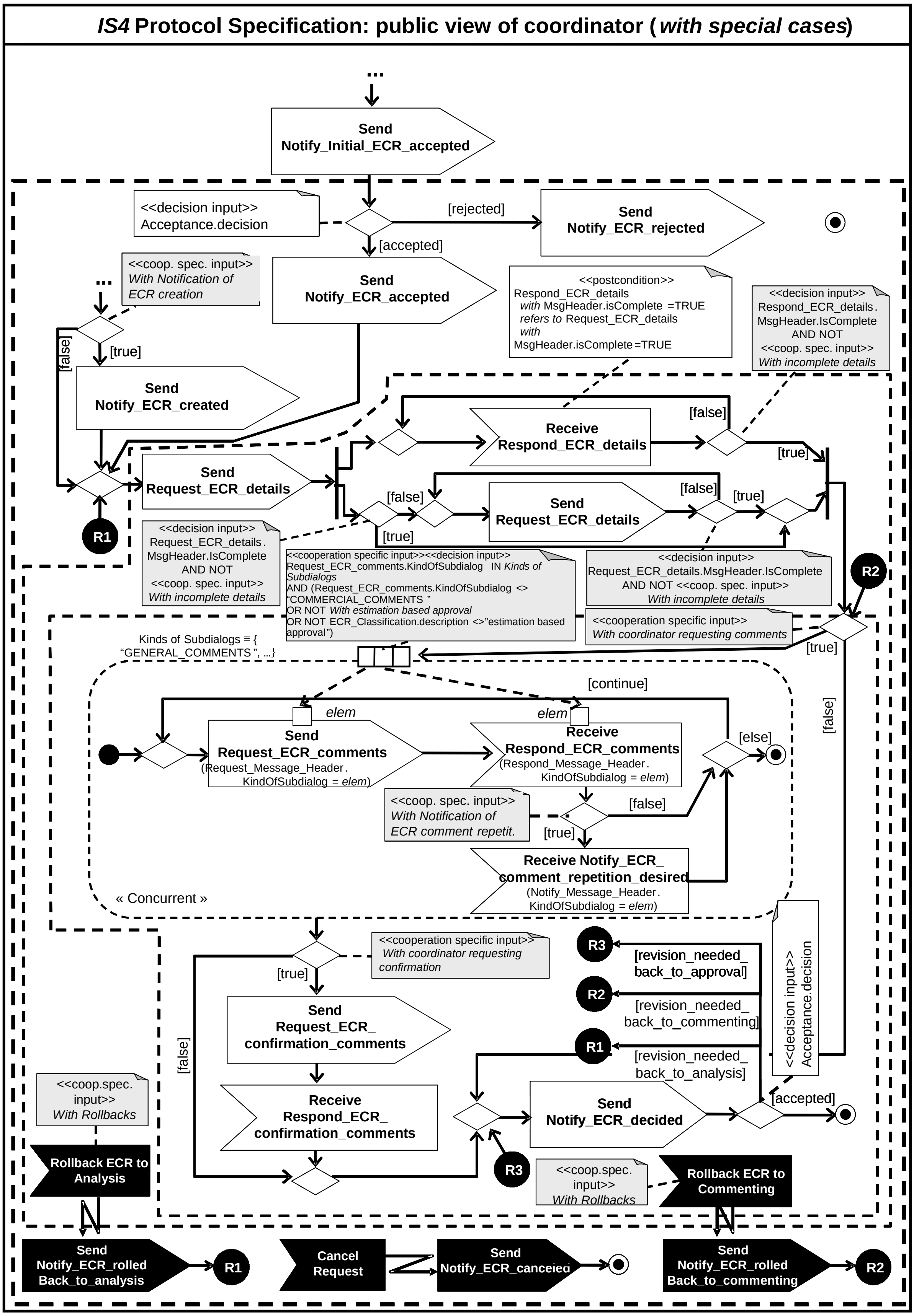

Figure 44: IS4 Protocol Specification: public view of coordinator (with special cases)

***IS4* Protocol Specification: public view of participant (*with special cases*)**

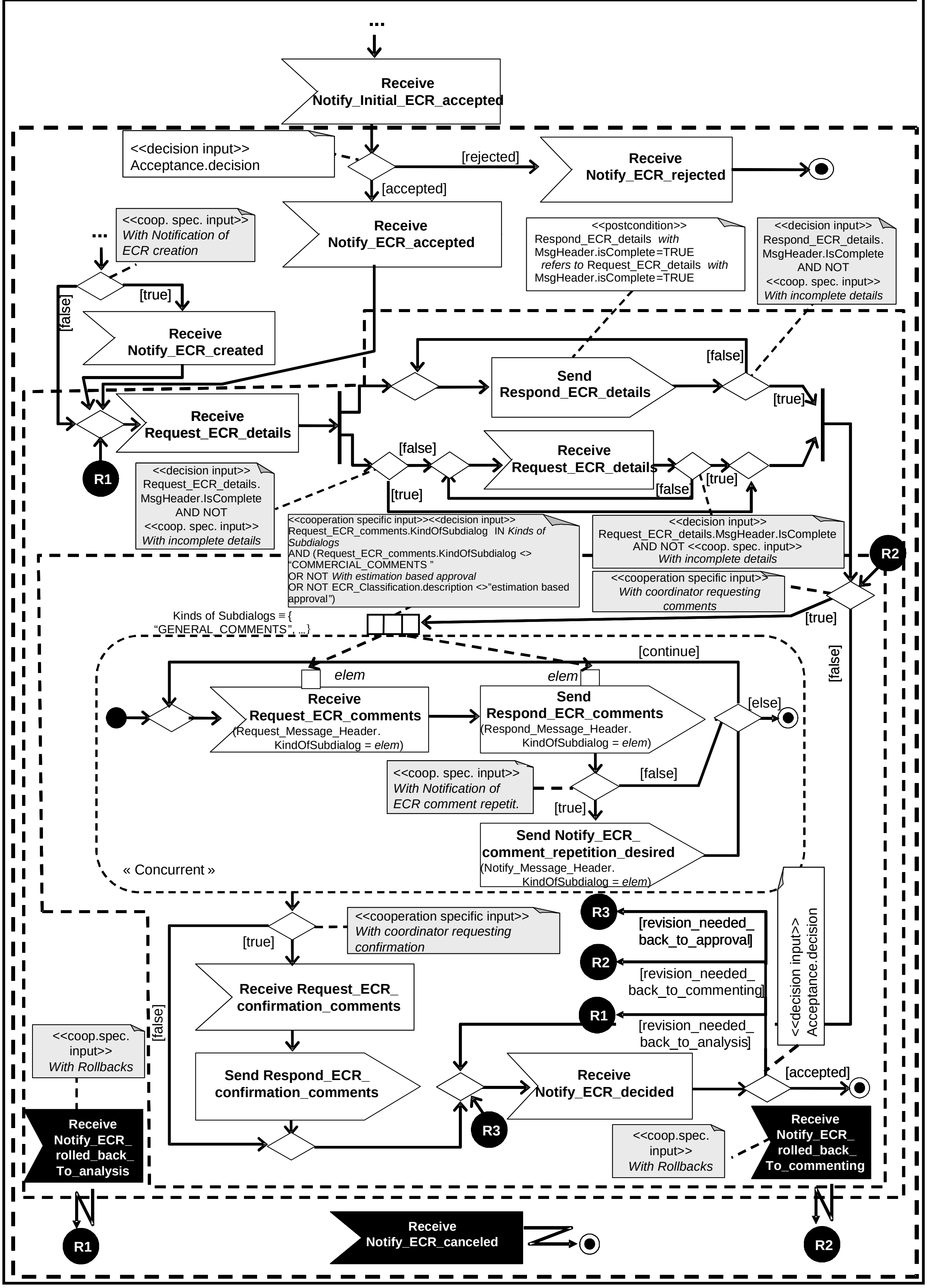

Figure 45: IS4 Protocol Specification: public view of participant (with special cases)

Maintenance Form

If you find an error or other changes that should be made to this publication, please complete this form and return it to the proper address below.

Name of Submitter:		Date:
Company:		
Company Address:		
Phone:	Fax:	E-mail:
CHANGE REQUEST (Use additional sheets if necessary)		
Page Number of Change:		
Document Currently Reads:		
Recommended Changes/Should Read:		
Recommended Additions:		
Reason for Change:		
Signature of Submitter:		

Submit this Change Request by email to ecm-issue@sasig.com or to your organization at one of the addresses listed below:

Automotive Industry Action Group
26200 Lahser Road, Suite 200
Southfield, MI 48034
USA
Phone: (248) 358-3570
Fax: (248) 358-3253
Web: www.aiag.org

Odette International Limited
Forbes House
Halkin Street
London SW1X 7DS
UK
Phone: +44 207 344 0277
Fax: +44 207 245 6093
Web: www.odette.org

Japan Automobile Manufacturer's Association, Inc. (JAMA)
Otemachi Building, 6-1
Otemachi 1-chome Chiyoda-ku
Tokyo 100-0004
Japan
Phone: +81 3-5405-6130
Fax: +81 3-5405-6136
Web: www.jama.or.jp

ProSTEP iViP Association
Dolivostr. 11
64293 Darmstadt
Germany
Phone: +49 6151 9287-336
Fax: +49 6151 9287-326
Web: www.prostep.org