Data Model and Tool Support for a consistent Functional Verification Chain in Space Projects

SEPS 2012 , ESTEC

All the space you need

esa

ASTRIUM
AN EADS COMPANY

Motivation & Context

esa

ASTRIUM
AN EADS COMPANY

What is it all about ?

- This talk is about a ongoing ESA study for Automation of Space System Test Data Collection, Processing and Reporting
- The main idea is to setup, fill and maintain databases, which store all information related to functional verification.
- This databases allow to auto-generate:
  - Test Plan, Test Specs, Test Procedures, Test Reports, Test Analysis Reports, ...

  and support corresponding consistency checks
- Further connections exist to: NCRs, Rfws, Operational Constraints, Configuration Control etc.
- The detailed results can not be presented in half an hour, but the reports can be obtained via ESA
- This talk introduces the overall study approach and what information can be found in the reports

Why could it be interesting ?

- Functional Verification is getting more and more complex, driving project schedule and cost. It is today's main challenge from project managers perspective.
- Very positive experience with FV Database driven approach in SWARM project (Dr.St.von der Nüll)
- The Functional Verification Chain is related to Tool-Boxes of future European Common Core Check-Out System **EGS-CC**

Study Content & Logic

Study Content & Logic

- **Task 1**: Identify and Document Processes and Roles of Functional Verification
- **Task 2**: Develop a Conceptual Data Model of Functional Verification
- **Task 3**: Specification of Informatics Tools to support the Process
- **Task 4**: Implications for current Processes and Tools
- **Task 5**: Implementation of a Demo

The presentation will focus on Task1 and 2

The approach was:

- usage of defined and clear notations
- as simple as possible in order to support discussions with domain experts rather than S/W or modeling experts

Task 1

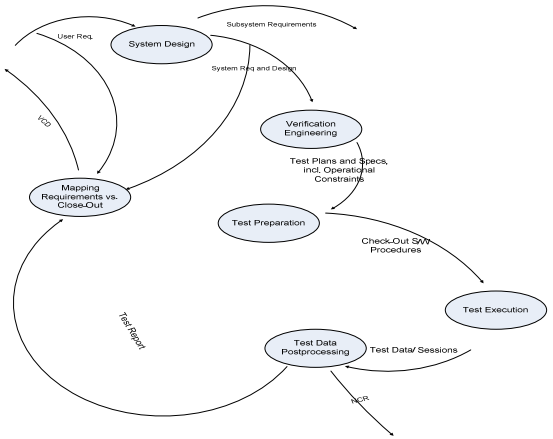
Identify and Document Processes and Roles of Functional Verification

esa

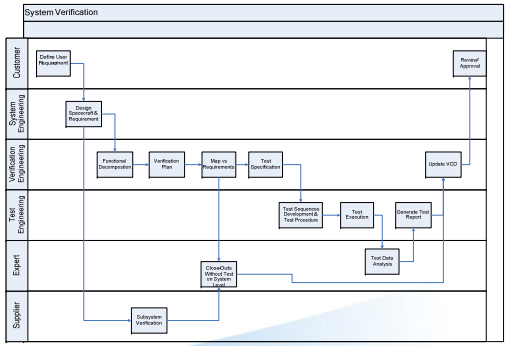
ASTRIUM
AN EADS COMPANY

Task 1 Logic

- Two Step Approach:
  - 1. Dataflow Analysis

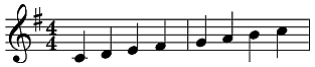


  - 2. Process Analysis and Roles



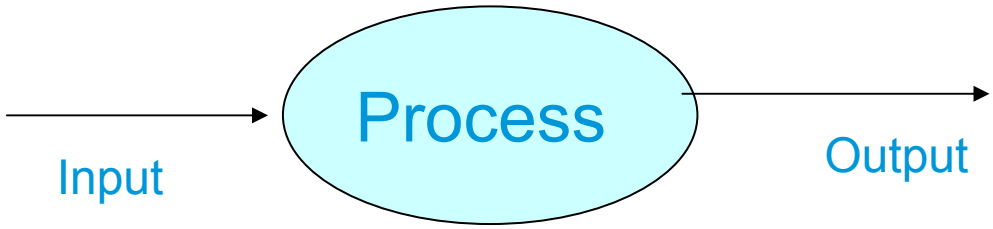
Task 1 Step 1 : Dataflow Analysis

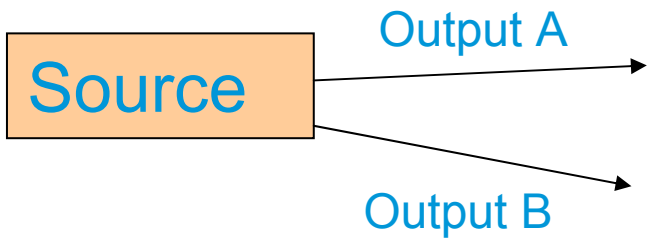
Task 1 Dataflow Analysis / **Notation**



Notation Used : DeMarco

- Source/Sink
- Process
- Dataflow

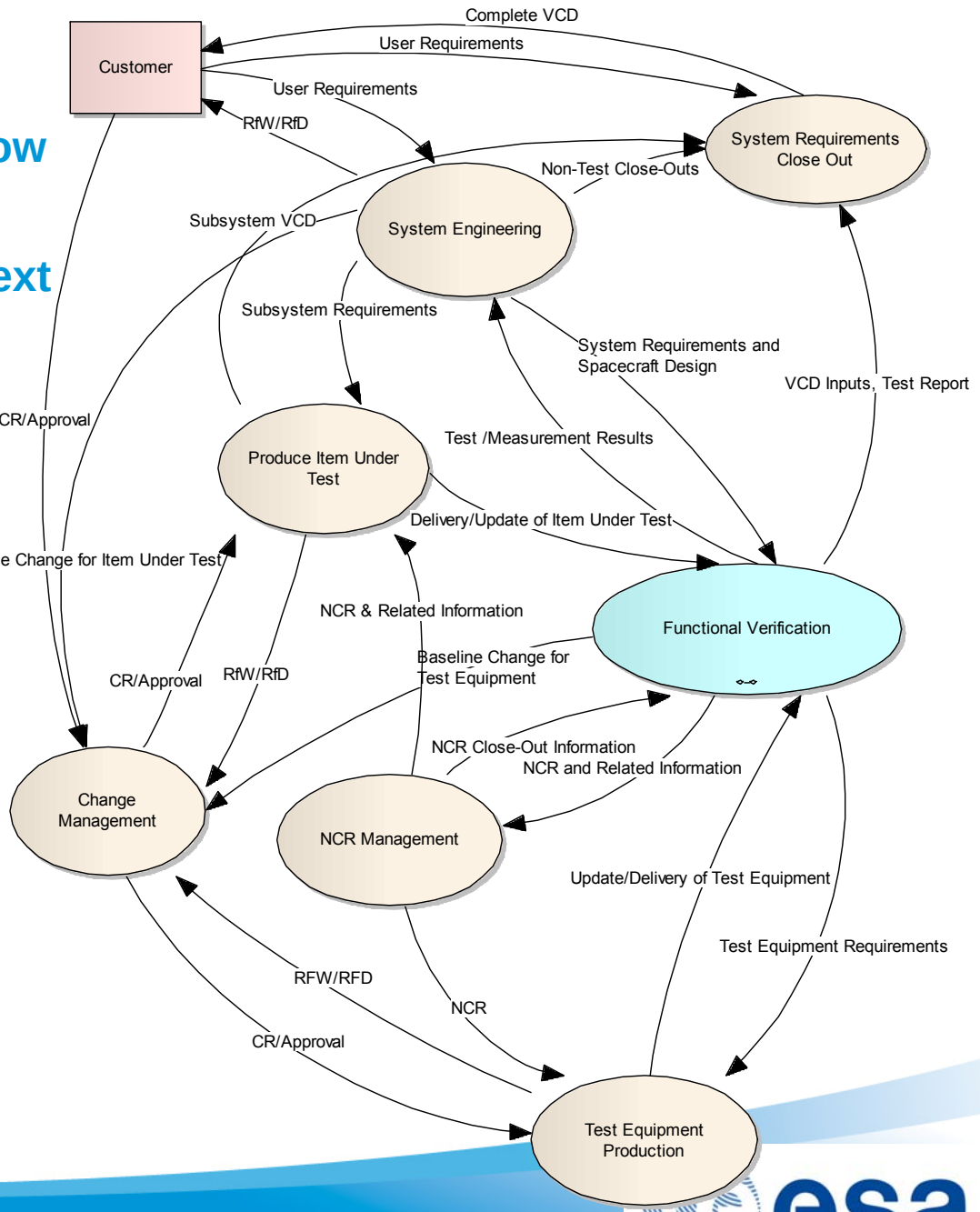




Dataflow Diagrams (DfD) show only Dataflows,
no time sequence and no conditions/branching/loops etc.

Task 1 Dataflow Analysis /

Level 0 Context



Task 1/ Dataflow Analysis

The diagrams might look crowded,
but in fact we have even more dataflows !

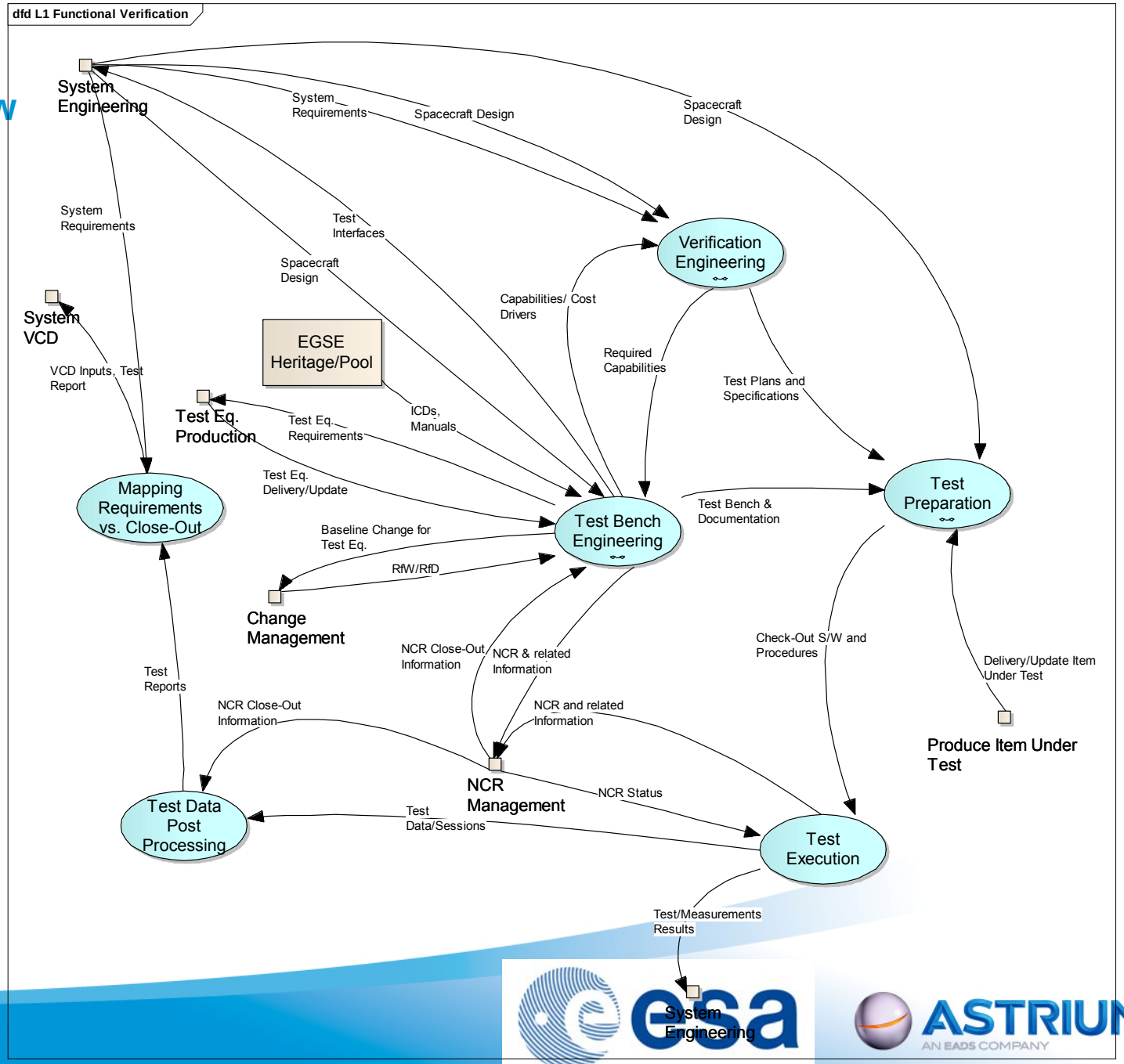
Implicit Dataflows not shown are:

- Feedback
- Review/Approval
- Configuration Control
- Schedule Monitoring

esa

ASTRIUM
AN EADS COMPANY

Task 1 Dataflow Analysis / Level 1 Functional Verification

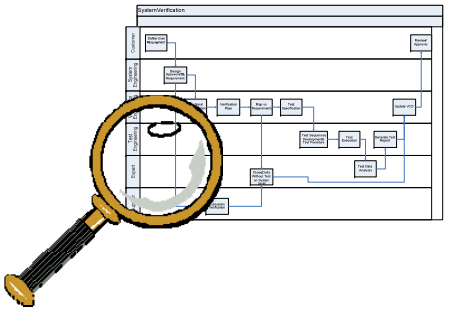


Task 1 Dataflow Analysis con ‘t

- Wherever a infinity ∞ symbol is shown in the diagram is refined to lower level.
- The following DfDs have been documented in Task 1 Report
  - Context Diagram (External I/F)
  - Verification Engineering
  - Test Bench Engineering
  - Test Preparation
  - Configuration Control
  - Configuration Control S/W
  - Configuration Control H/W

Task **1** Step **2** : Process Analysis and Roles

Task 1 Step 2 / Process Analysis



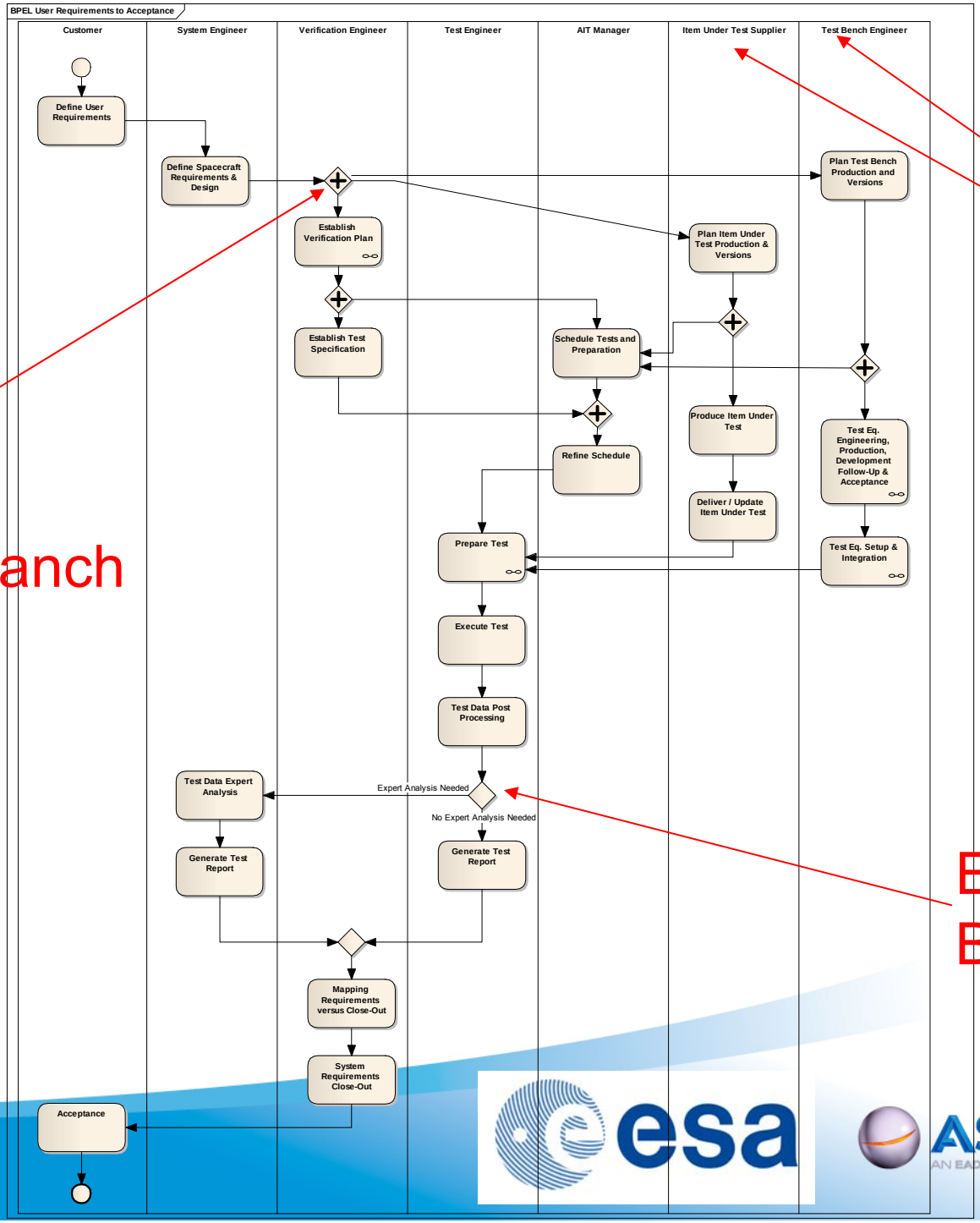
- Shows the sequential order of tasks
- Defines the roles
- Show the decision points of the sequence

- DOES NOT SHOW THE DATAFLOWS

Task 1 Process Analysis /Notation

Level 0

Requirements to Close-out

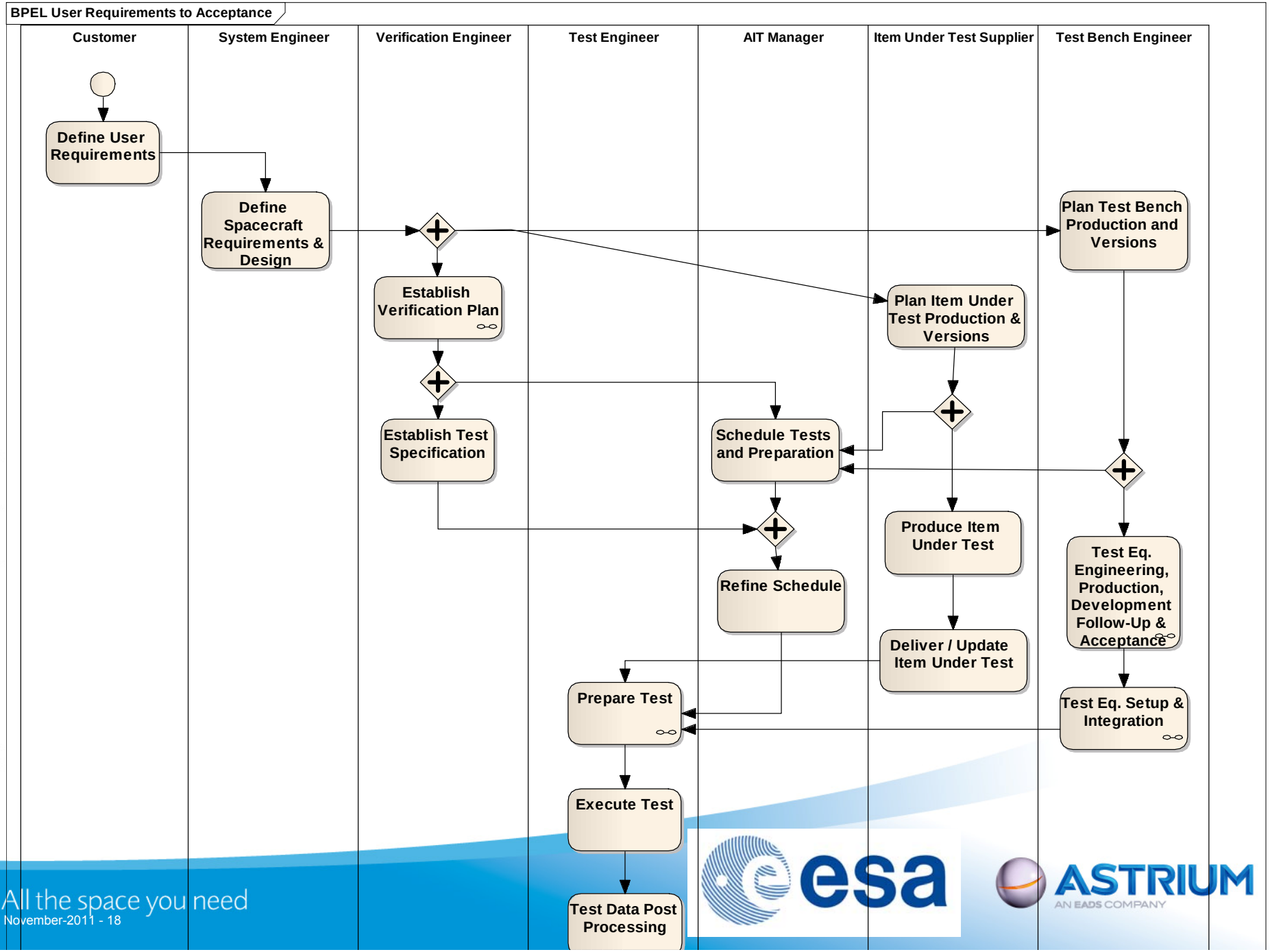


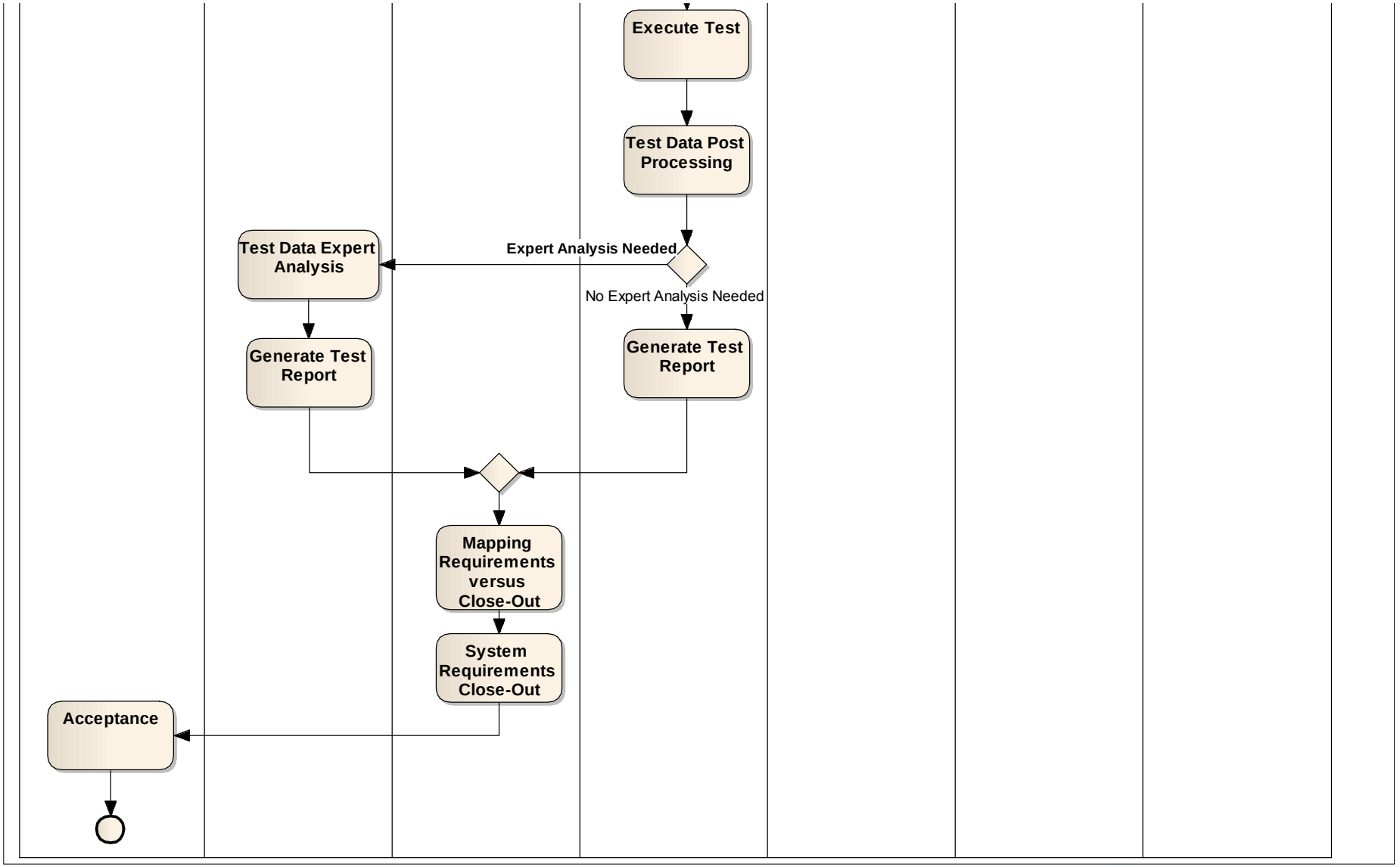
Role

Parallel Branch

Time

Excl. Branch





Task 1: Process Analysis

- The following BPNM diagrams have been generated:
  - **User Requirements to Acceptance**
  - **Establish Verification Plan**
  - **Establish Verification Plan (Req. Driven)**
  - **Test Eq. Engineering, Production, Development Follow-Up & Acceptance**
  - **Test Eq. Setup & Integration**
  - **Test Preparation**
  - **NCR Process**
  - **RfW**
  - **Generic: Baseline Change Process**
  - **Generic: Concurrent Engineering**
  - **Generic: Hierarchical Coordination**
  - **Generic: Feedback (Formal Review)**
  - **Generic: Feedback (Informal & Continuous)**

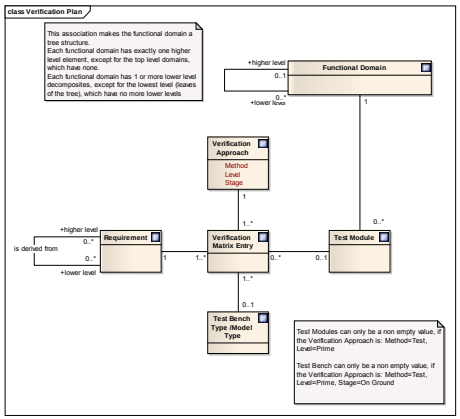
Task 2

Develop a Conceptual Data Model of Functional Verification

esa

ASTRIUM
AN EADS COMPANY

Task 2 Logic

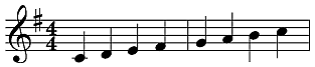


1. Conceptual Data Model on Entity Level, get the Multiplicities right

2. Discussion of Versioning / Config Control of the Database itself

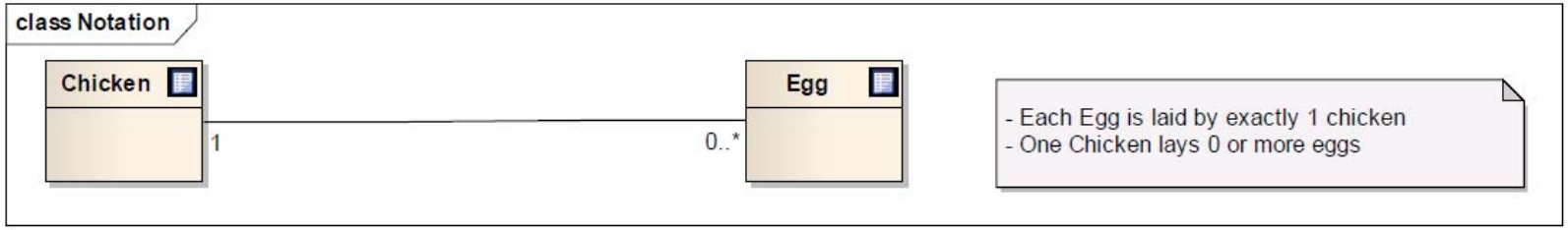
3. Detailed Data Content of the Entities

Task 2 / **Notation**

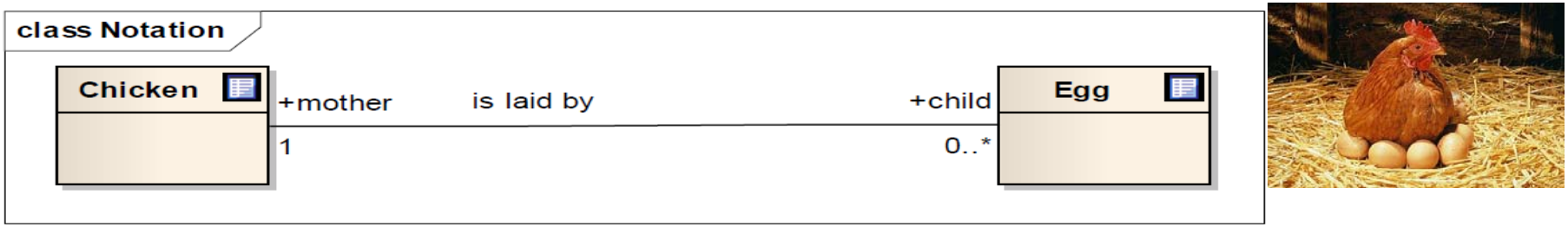


- UML used in ORM / ERM fashion (simply because EA only supports UML here), emphasis on the cardinality/multiplicity

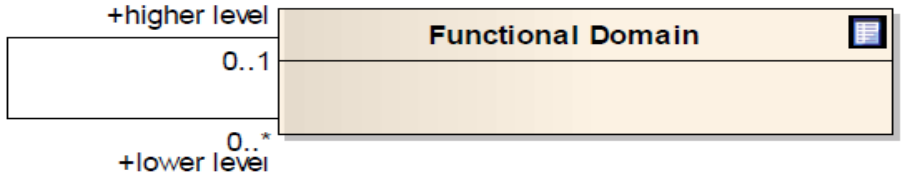
- basic association style



- roles & named associations



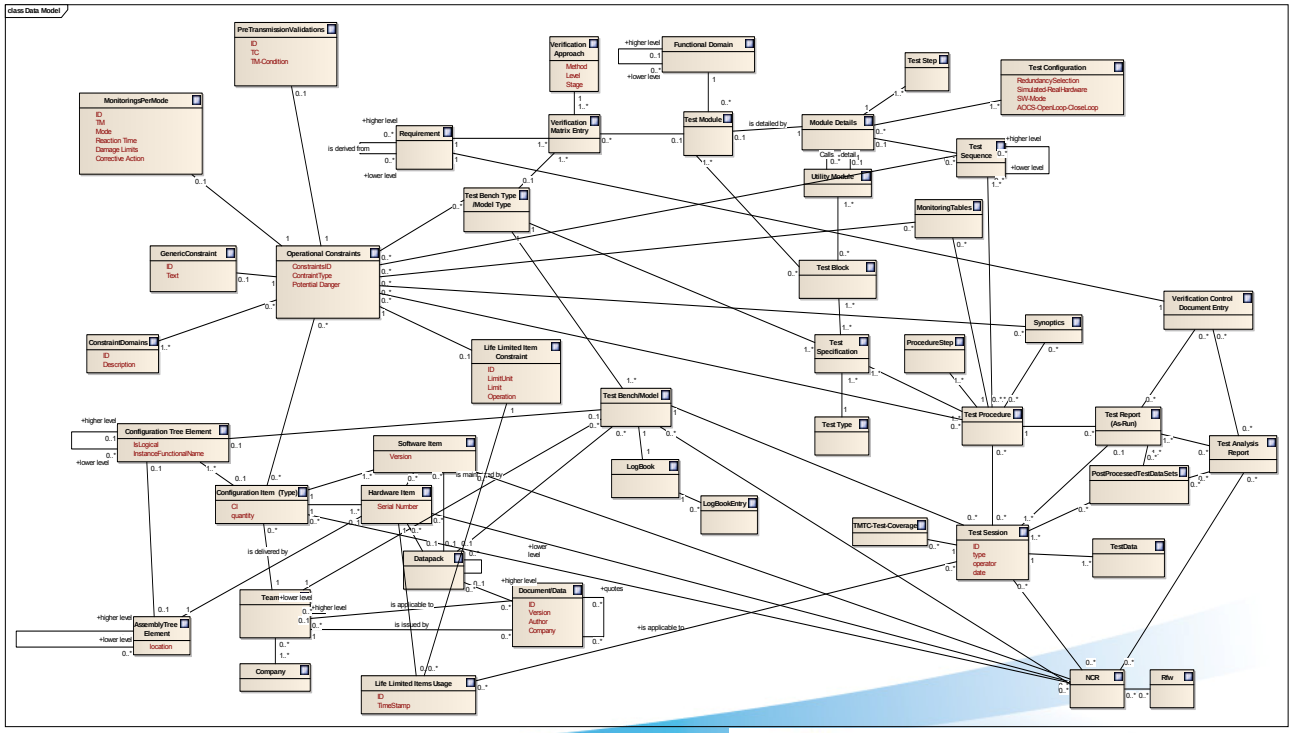
- recursive associations



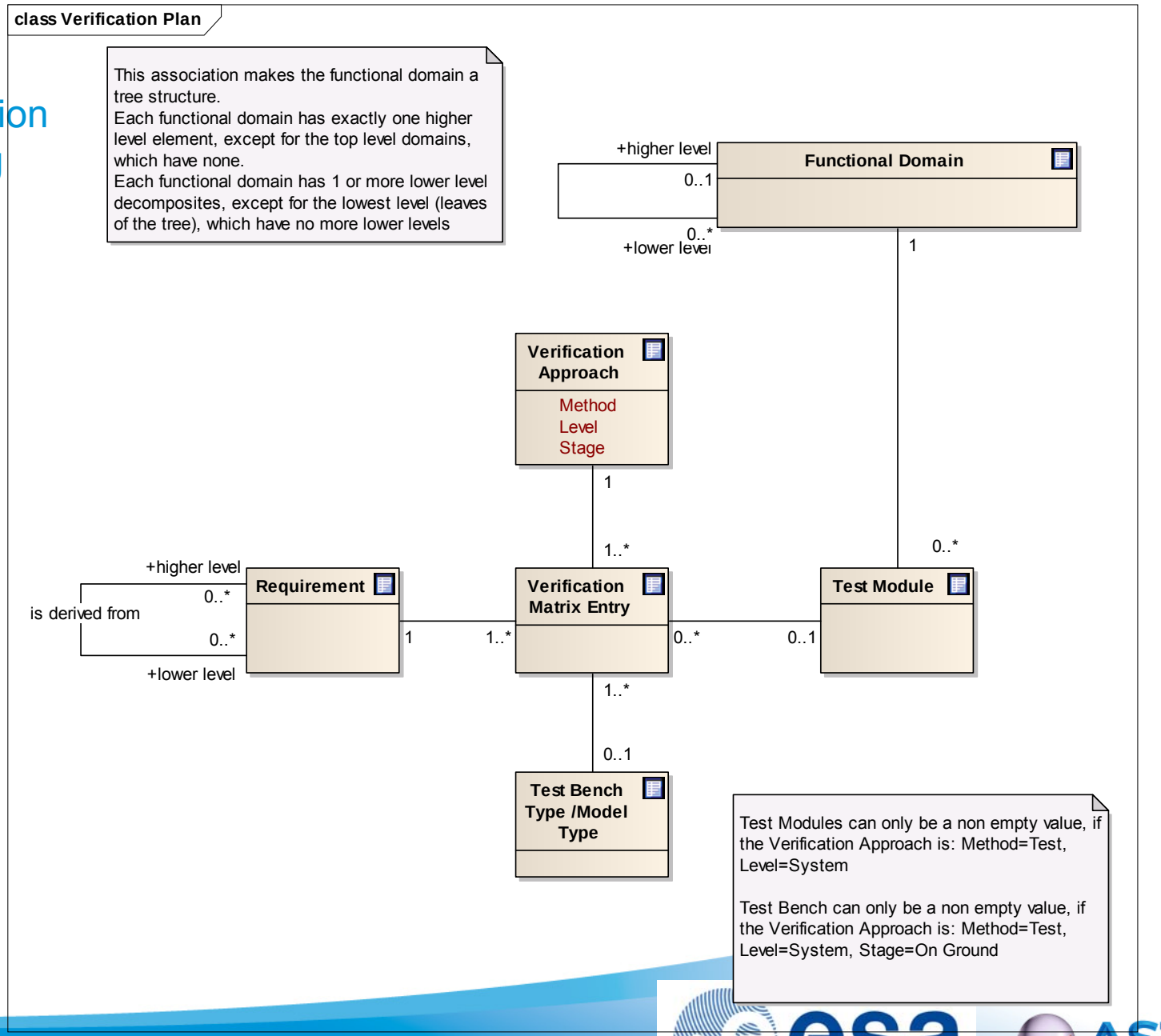
This association makes the functional domain a tree structure.
Each functional domain has exactly one higher level element, except for the top level domains, which have none.
Each functional domain has 1 or more lower level decomposites, except for the lowest level (leaves of the tree), which have no more lower levels

The Complexity Overall Datamodel

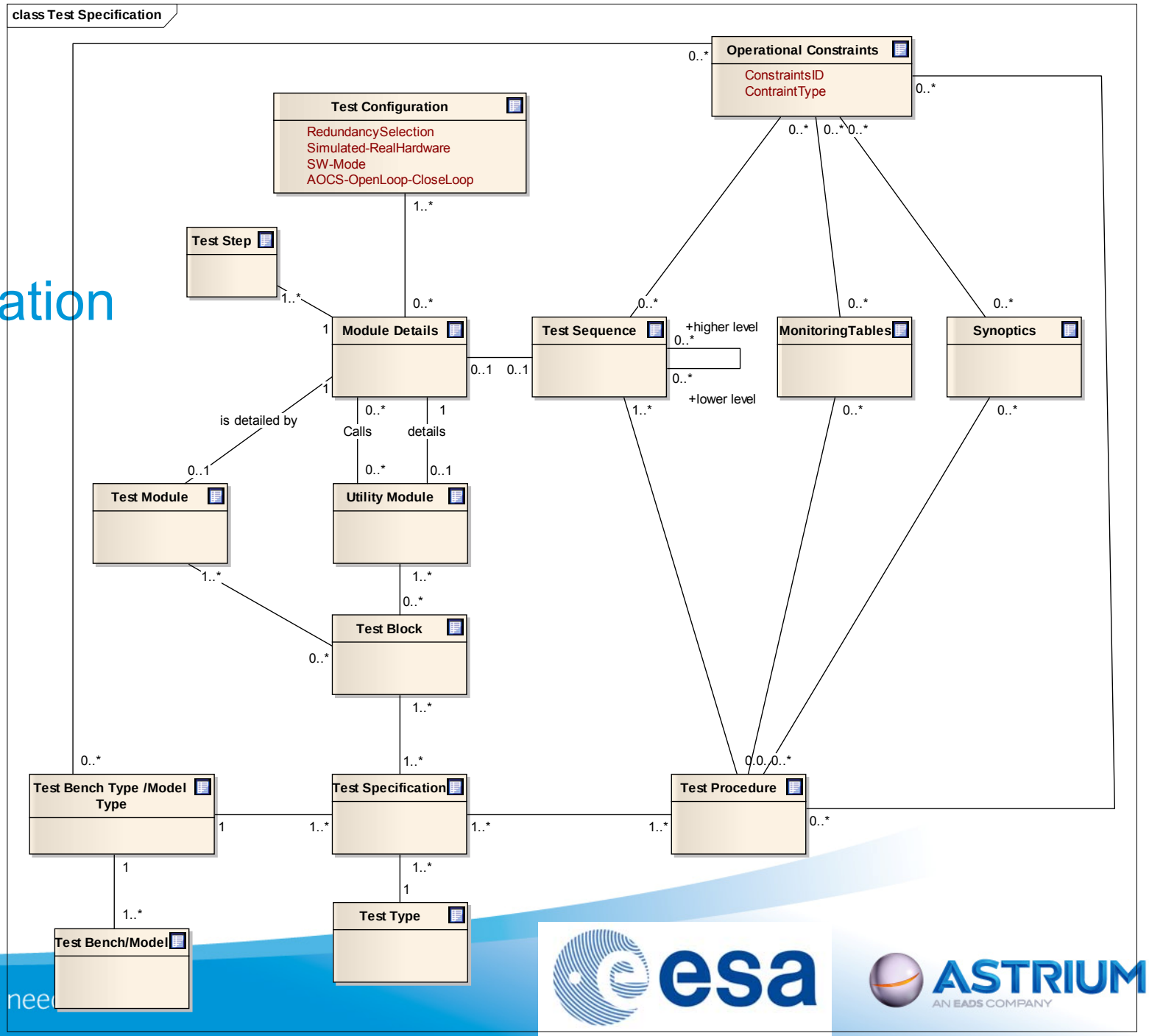
- 46 entities (for comparison: SCOS2000 has about 50 tables)
- 80 associations, mostly many-to-many
  - ⇒small/mid size database S/W project
  - ⇒graphical user interface and consistency checks are mandatory



Task 2 / Verification planning



Task 2 / Test Specification



esa

ASTRIUM
AN EADS COMPANY

Task 2 / Conceptual Data Model

The following Views on the Conceptual Data Model have been generated:

- Verification Plan
- Test Specification
- Test Preparation & Execution
- Verification Control
- Operational Constraints
- Configuration Control

For each entity (Table) also the detailed attributes (columns) have been worked out

Task 2 Versioning

V 1.0 V1.1 V2.0 V2.0B

Motivation

- All data is directly or indirectly connected
- Updates of late phases data
  (e.g. Test Reports) must not trigger Re-review
  of early phases data (e.g. Verification Plan or Test Spec)
- It must possible to check consistency of early phase approved baselines with late phase data.
- Different Teams can be on different baselines

⇒Freezing the whole database at milestones and giving this one version name will not be sufficient

- The model will contain thousands of small pieces of information

⇒A different independent version for everything will not work either

esa

ASTRIUM
AN EADS COMPANY

Task 2 Versioning

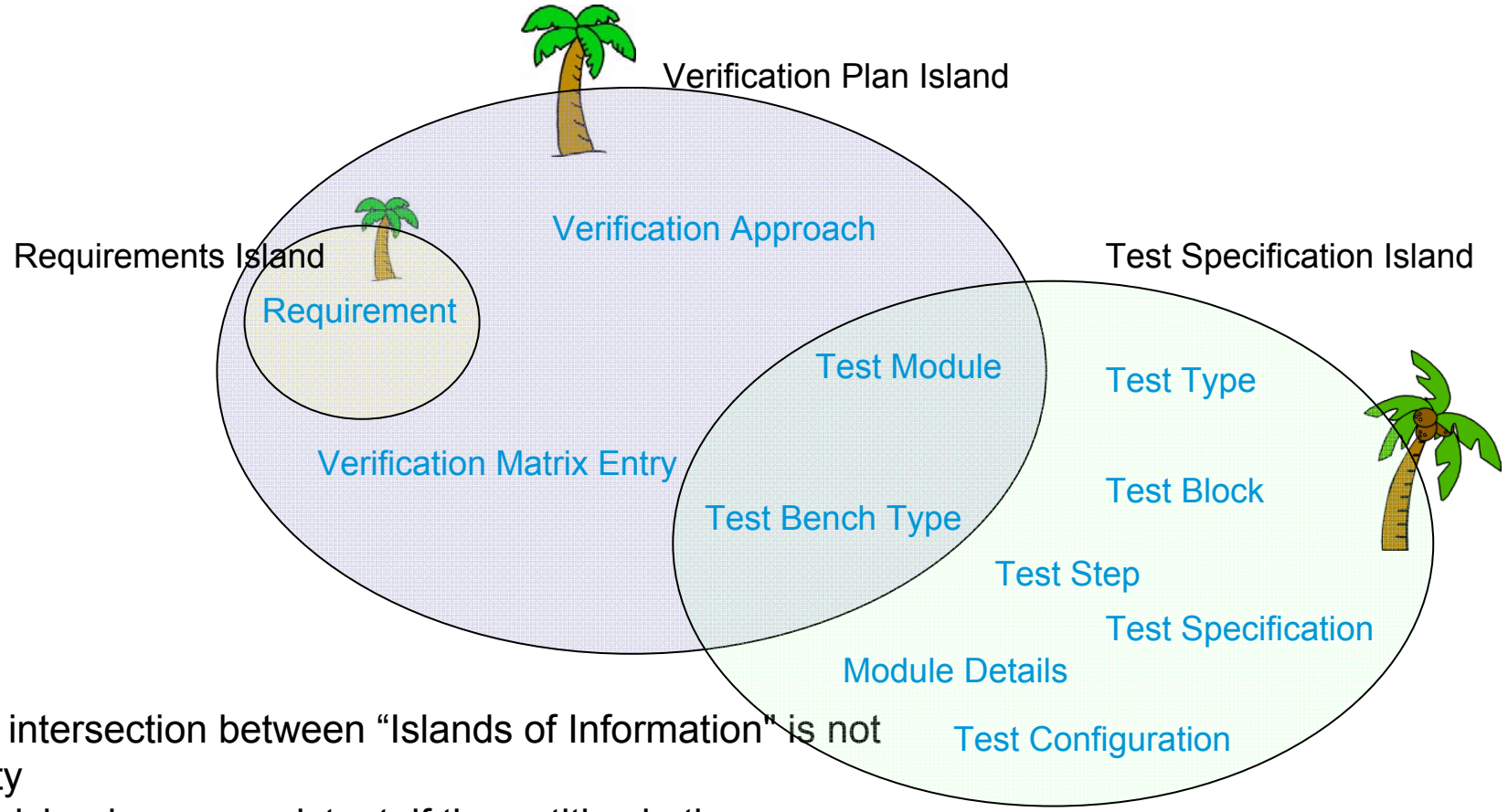
V 1.0 V1.1 V2.0 V2.0B

- Therefore a feature is needed allowing to assign versions to data subsets of the database. These subsets are called “islands of information” in the study.

Definition:
An “island of information” is a set of objects (rows in tables) serving a common purpose and hence requiring a separate version control.

Example: All Data which goes into the Test Specification of AOCS Normal Mode Testing.

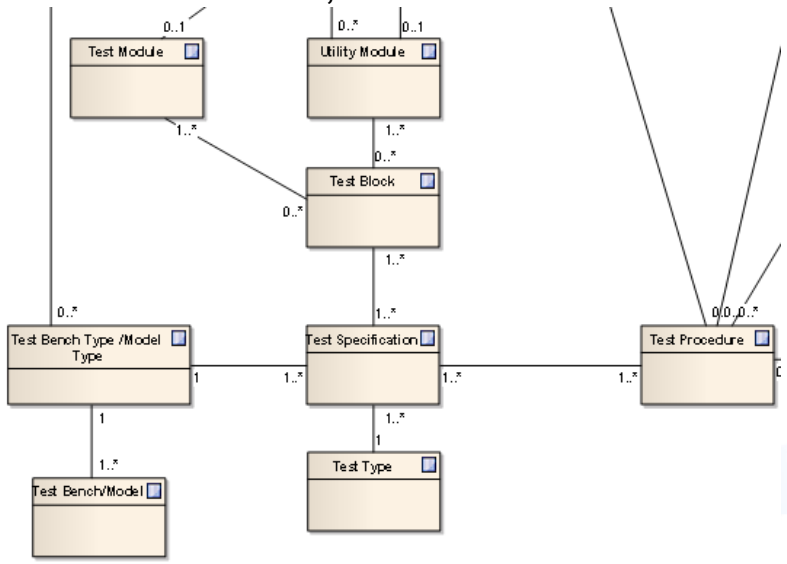
Task 2 / Island of Information Concept



- The intersection between “Islands of Information” is not empty
- Two islands are consistent, if the entities in the intersection have the same version
- No “offshore-association”, i.e. between entities of different islands, if not in the intersection

Task 2 / “Island of Information” Concept

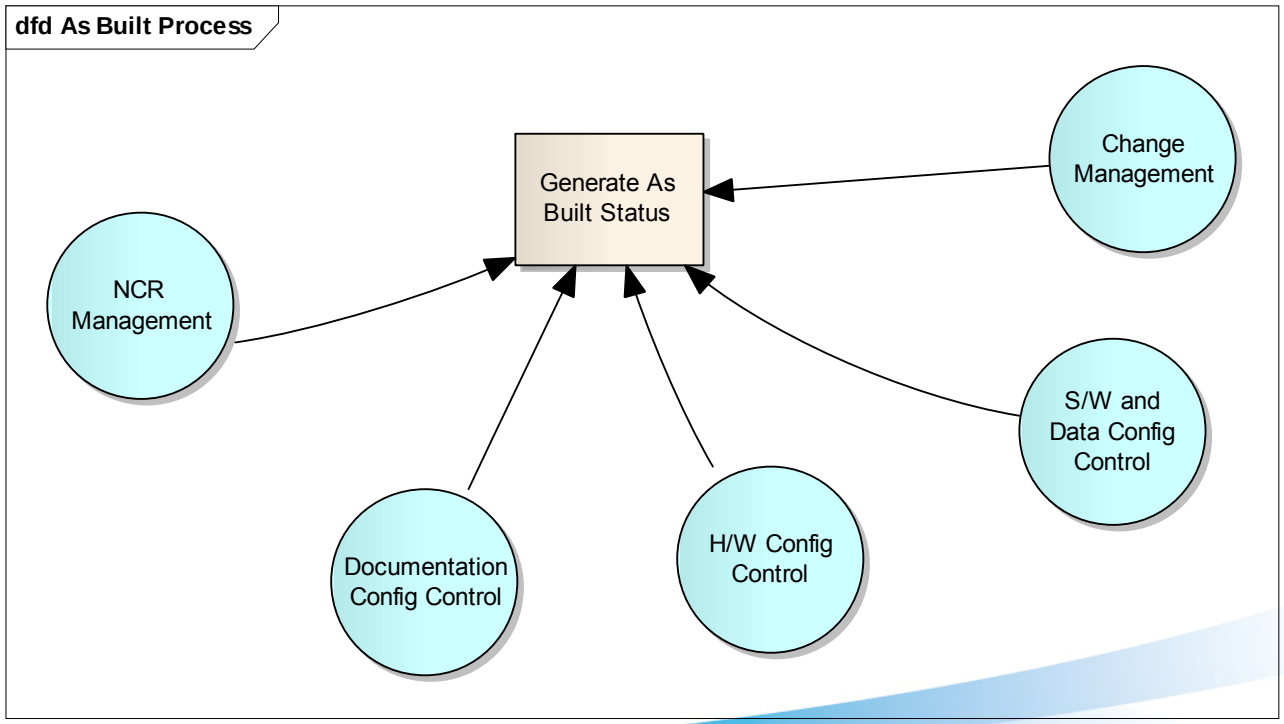
- Two different Types of Islands exist:
  - Type 1 (Simple) :
    All information of all entities (or Table-Entries) is tagged with the same TAG, Example: Baselines in DOORS, all requirements receive the TAG “CDR_VERSION”
  - Type 2 (Deep Search Type):
    All information related to one entity is tagged with the same TAG, Example: Test Specification Island, TAG AOCS_TSPE_V2_0.



Backup Slides

Task 1 Dataflow Analysis / Configuration Control

- "As Built" is typically done manually or semi-automated
- Intermediate Layers of config control (e.g. Datapack, S/W CIDL, ...)
- Note: There could be benches using virtual machines (Hardware changing ...)



Task 1 Dataflow Analysis / Configuration Control

- S/W and Databases:
  - Come in versions
  - Can be used/installed @ various locations @ the same time
  - High update frequency during development, without formal review cycle (for check/out S/W up to several updates per day)
  - typically two stage config control (“day by day” + formal milestones)

- Documentation
  - Comes in versions
  - Can be used/installed @ various locations @ the same time
  - Moderate Update Frequency
  - Applicability (different versions applicable to different teams)
  - Approval Status
  - If Databases replace documents, applicability becomes relevant for databases

- H/W
  - Unique with CI + Serial Number
  - No versions
  - Low update frequency, small material comes in batches/lots

esa

ASTRIUM
AN EADS COMPANY

Task 1 Dataflow Analysis / Configuration Control

- NCR Management:
  - like Documentation, but
  - has Status
  - is linked to configuration controlled items (S/W, H/W, Document)

- Change Management
  - like NCR management
  - also linked to contract

Task 1 Process Analysis /

Notation BPMN

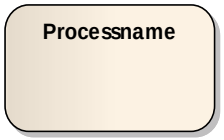
BPEL Business Process Model Notation

Start

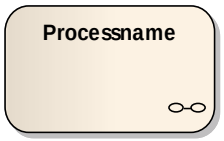
End

Gateway: Here flows branch or come together. Without symbol inside it is an exclusive gateway, i.e. only one branch can be taken.

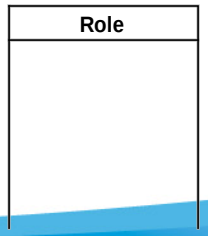
Gateway: Here flows branch or come together. With the + symbol inside it is an parallel gateway, i.e. branches will be exercised in parallel



Process



Process, which is further decomposed into sub processes in a lower level diagram. This is indicated by the infinity symbol

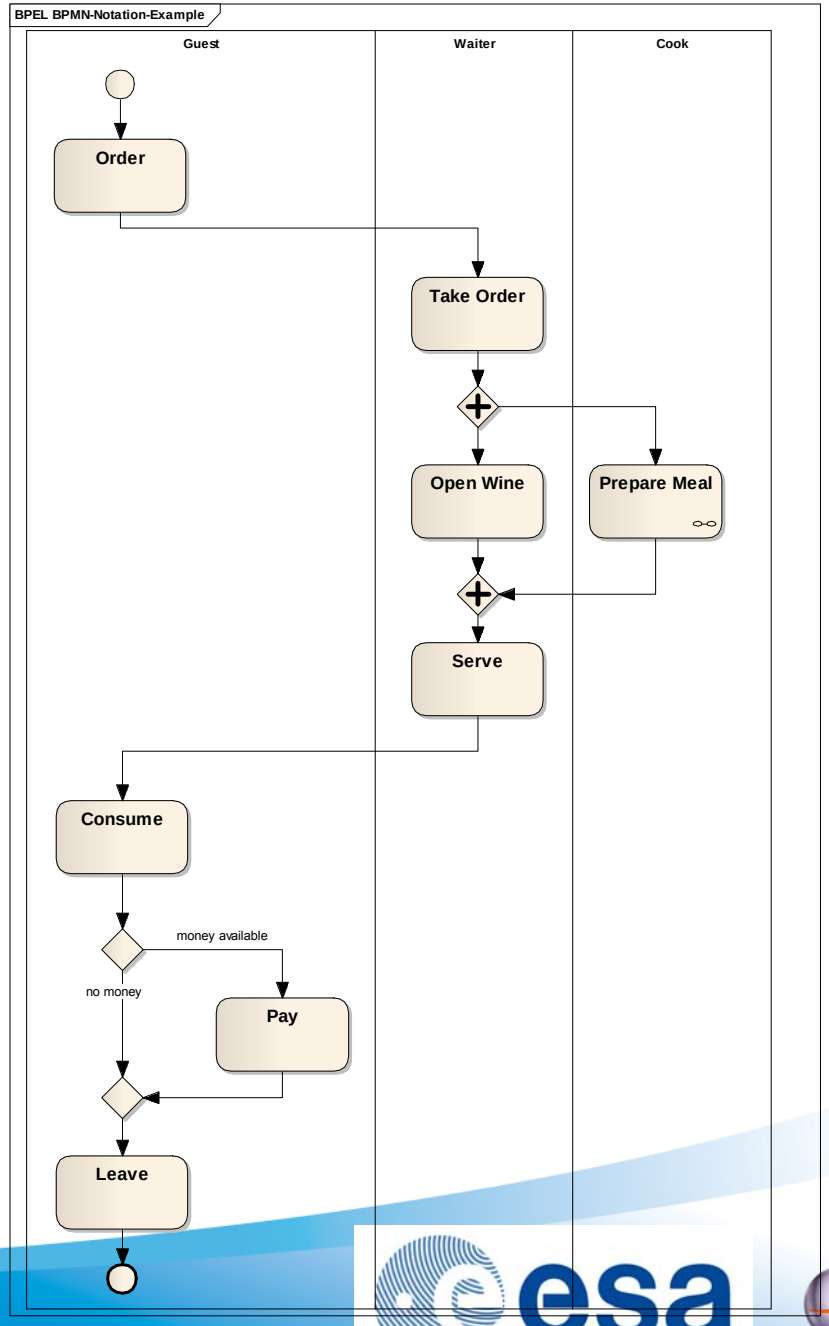


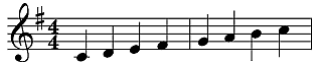
Swimlane: All processes inside a swimlane will be performed by the Role indicated

Task 1 Process Analysis /

Notation BPMN

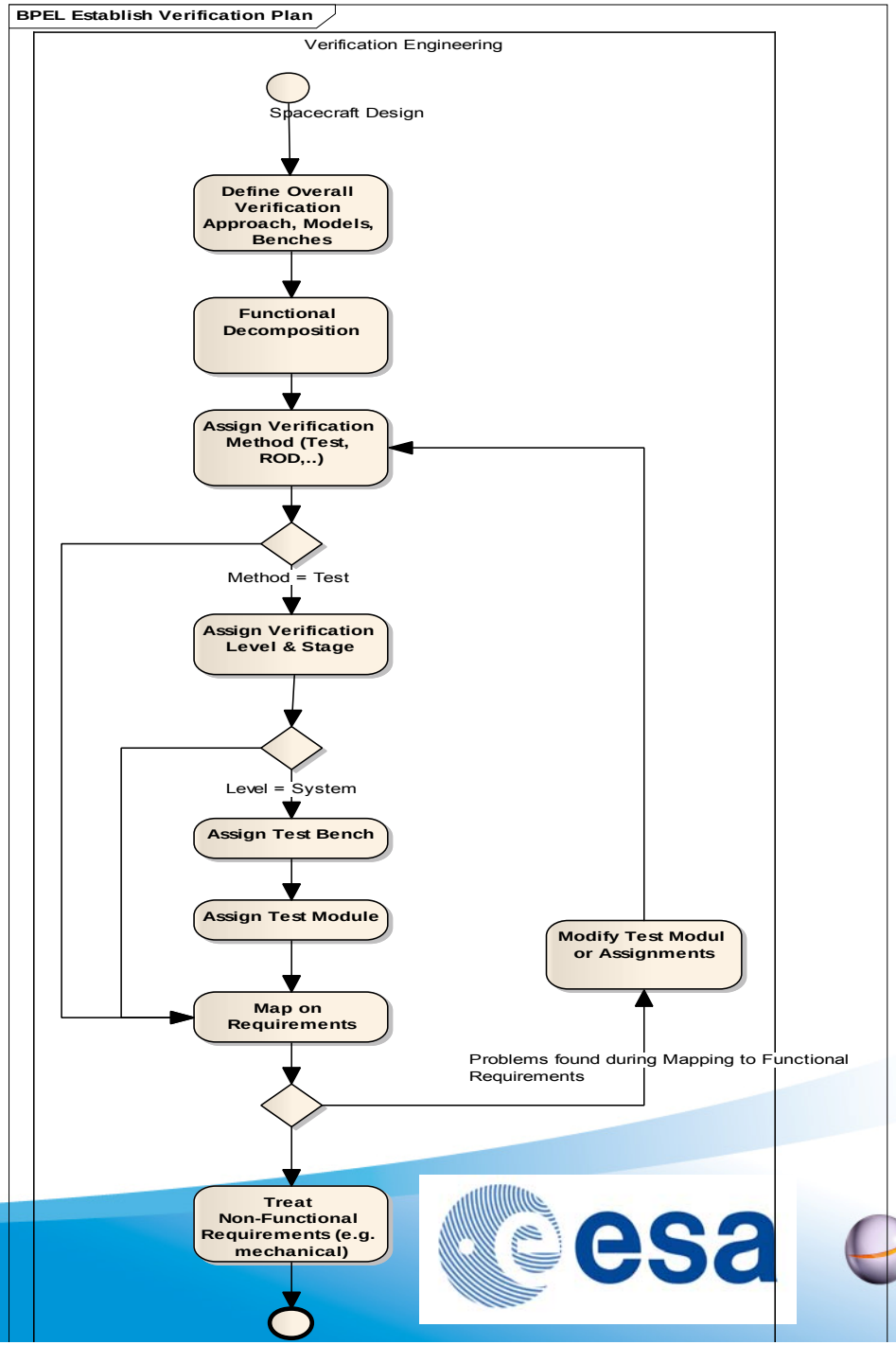
Example: Restaurant





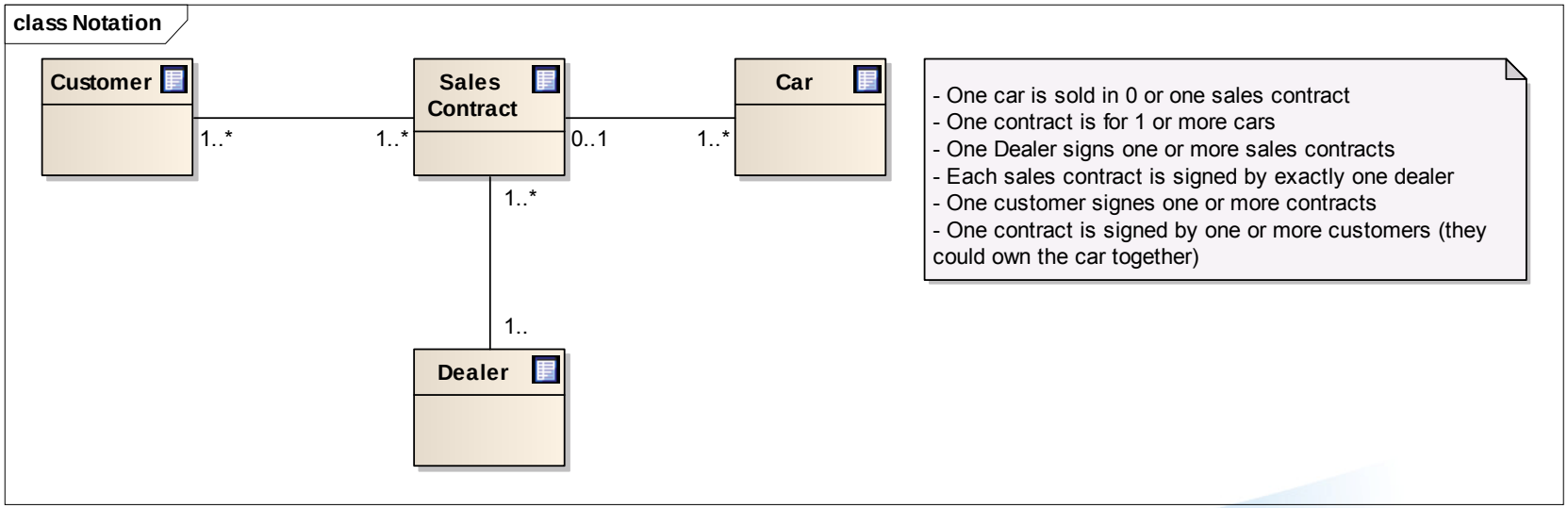
Level 1:

Establish Verification Plan



and many more ...

Task 2 / **Notation** (n-ary relations)



Task 2 / “Island of Information” URD

- a) Whenever a data entry in one of the data entities is saved and released for used by others the entry shall receive a new version number (e.g. a "Module Detail" Entity is updated, is gets a new version. Test engineers need to update the corresponding test sequence)
- b) For user ergonomics it needs also to be possible to release all entries of an entity. Only the modified ones shall be incremented wrt. version in this case (E.g. release all "module details", all versions of "module details" modified since the last release are incremented).
- b) Beside from the versions related to single data entries, it needs to be possible to tag "islands of information" with a common version name. (E.g. all data related to AOCS ISST test specification is tagged AOCS_ISST_SPEC_2B). The natural "islands of information" follow the deliverables of the ECSS (verification plan, test specification, test procedure, …).
- c) The intersection between "island of information" is not empty. E.g. a Test Module is part of the "verification planning island" and the "test specification island". Example: the Test Module "GPS Switch Over" has the version 5.27, it also has the tag VERIFICATION_PLAN_5A and the tag AOCS_ISST_SPEC_2B.
- d) Versioning needs to support the capability to check, if given versions of two different "islands of information" are consistent. This is done by checking, if all data being common to the different "islands of information" is of the same version. Example: Check if AOCS_ISST_SPEC_2B is consistent to VERIFICATIO_PLAN_5A (Example for the outcome of such a consistency check could be: Everything O.K., but the Test Bench proposed by the VERIFICATION_PLAN_5A for the Test Module "GPS Switch Over" is EFM, while the AOCS_ISST_2B defines SVF as the Test Bench to be used.)
- e) Last not least versioning also needs to support that all data stored the data model is assigned a common tag (e.g. STATUS_SATELLITE_CDR, or BACKUP_BEFORE_CHRISTMAS).