Functional Specification of CP23

Rieks Joosten

29 May 2014

Contents

Chapter 1

Introduction

This document[1] defines the functionality of an information system called 'CP23'. It defines the database and the business services of CP23 by means of business rules[2]. Those rules are listed in chapter 2, ordered by theme. , ordered by theme.

The diagnosis in chapter 3 is meant to help the authors identify shortcomings in their Ampersand script.

[1] This document was generated at 29-5-2014 on 21:41:42, using Ampersand v3.0.2.1354, build time: 29-May-14 05:59:43 UTC.

[2] Rule based design characterizes the Ampersand approach, which has been used to produce this document.

Chapter 2

Shared Language

This chapter defines the natural language, in which functional requirements of ‘CP23’ can be discussed and expressed. The purpose of this chapter is to create shared understanding among stakeholders. The language of ‘CP23’ consists of concepts and basic sentences. All functional requirements are expressed in these terms. When stakeholders can agree upon this language, at least within the scope of ‘CP23’, they share precisely enough language to have meaningful discussions about functional requirements. All definitions have been numbered for the sake of traceability.

2.1 Policy 2.3 vsn 1: Asset Management - Portable Equipment

Company Inc. has concerns regarding equipment that employees can ‘carry around’, such as cell-phones, laptops, cars, toolboxes, ID-cards, etc. In order to address these concerns, this policy specifies rules for the purpose of achieving the following objectives: # employees must dispose of all company equipment that is necessary for doing their jobs. # total cost of ownership of company equipment must be controlled, which includes costs for stocks and usage/license fees, in particular when equipment is not or no longer in use. # risks associated with company equipment must be at an acceptable level, not just for company-owned equipment, but also for equipment owned by employees themselves. This pattern defines the agreements necessary to follow the rules that aim to achieve these objectives.

At this point, the definitions of *employee*, *organizationalRole*, *equipment*, *employeeName*, and *eqtKind* are given.

In order to distinguish between people that work for Company Inc. and those that are not, we define the term ’Employee’.

Definition 1: a person that has been issued a personal ID-card of Company Inc. [CP2.3v1:3.1] *Employee*

In order to refer to employees within the HRM system in a way that is recognizable by people as well, we need each employee to be assigned a unique name.

Definition 2: a human readable text that uniquely identifies an employee *EmployeeName*

Within Company Inc., responsibilities are grouped in sets that indicate what kind of work is to be done, and that is meaningful to the organization. We introduce the term 'organizational role' to refer to such sets. Examples include 'HRM officer', 'Manager', 'Security Officer', 'Programmer', 'Salesperson'.

Definition 3: a set of (related) responsibilities as defined by Company Inc., assigned to employees [CP2.3v1:3.3] *OrganizationalRole*

In order to express requirements for equipment that should be be assigned to employees, the kind of equipment must be identifyable. Examples include 'cell-phone', 'laptop', 'car'.

Definition 4: A class of equipment *EqtKind*

Employees need equipment to do their job, such as mobile phones, laptops, cars, toolboxes, ID-cards, etc. In order to keep track of such equipment, in particular when it is portable (moveable) it needs to be registered. Company Inc distinguishes between 'portable equipment', i.e. equipment owned by Company Inc. (P2.3:3.5)., and 'personal equipment', i.e. equipment owned by an employee of Company Inc. (P2.3:3.6).

Definition 5: an (identifiable) object that can be moved/taken away with relative ease, and that employees may need to do their job [CP2.3v1:3.4] *Equipment*

Within Company Inc. every employee has precisely one name, that identifies the employee. This allows the unambiguous registration of employees.

Agreement 6: Employees have a name

Phrases that can be made are for instance:

E10961 is referred to by ' Jean-Pierre Chanod '.

E20962 is referred to by ' Sean Alespy '.

E31423 is referred to by ' Thierry Jacquin '.

Employees may be issued non-standard equipment provided this is apporved by their manager. Therefore, the manager of employees must be known.

Agreement 7: Employees have been assigned a manager

The responsibilities that employees have are defined by the organizational roles that they fulfill. Depending on such roles, employees will be assigned standard issue equipment. *CP2.3v1:2.4*

Agreement 8: Employees have been assigned (at least) one organizational role that indicate(s) the kind of work they do

Phrases that can be made are for instance:

Jean-Pierre Chanod has been assigned the organizational role Director.

Jean-Pierre Chanod has been assigned the organizational role Employee.

Sean Alespy has been assigned the organizational role Salesperson.

Company Inc. has decided to issue equipment to employees based on their organizational role(s). Hence, for every organizational role, it must be possible to define the kinds of equipment that people in such a function must be assigned. This is the so-called the standard issue equipment for the organizational role. *CP2.3v1:3.7, 2.4*

Agreement 9: Employees that serve in an organizational role must be assigned equipment of specific kinds

Phrases that can be made are for instance:

Every employee in the role of Director must be issued a Computer.

Every employee in the role of Employee must be issued a Badge.

Every employee in the role of Employee must be issued a Phonenumber.

In order to keep good track of portable/mobile equipment that has been issued to employees, every equipment issued to an employee must be registered as such. *CP2.3v1:2.1, 2.4*

Agreement 10: An employee can be issued company equipment (for which it then is responsible)

Phrases that can be made are for instance:

Jean-Pierre Chanod has been issued CardMan CP1 304-682-231.

Jean-Pierre Chanod has been issued Vodafone Mobile 0693826586.

Jean-Pierre Chanod has been issued Nokia N32 407-21.

Employees are allowed to use personal equipment for their work, provided that they register such devices. *CP2.3v1:3.6*

Agreement 11: Employees may use their own portable equipment for their work

One employee may only manage another employee if he has been assigned the necessary responsibilities. Such responsibilities are defined for the role ’Manager’. Hence, employees may only be managed by (other) employees that fulfill this role. *CP2.3v1:3.2*

Agreement 12: An employee can only be managed by an employee that fulfills the organizational role of ‘Manager’.

Agreement 13: Issuing equipment pertains to company equipment only.

All personal equipment and company equipment that is issued to an employee, must be (implicitly or explicitly) approved for use by our Security Officer. *CP2.3v1:2.3-1a*

Agreement 14: Personal equipment and company equipment that is issued to an employee must have been approved for use.

2.2 Equipment

This theme defines the terminology that Company Inc. needs to address concerns related to (portable/mobile) equipment

Agreement 15: Equipment must either have a company ID or be owned by an employee

In order to do automated reasoning with statuses, and to ensure that statuses remain meaningful, it is necessary to control the allowed values for equipment statuses.

Agreement 16: Equipment may only be assigned a status ’Functional’, ’Not functional’ or ’Lost’

2.3 EquipmentIssuerProcess

This process specifies the responsibilities of an EquipmentIssuer related to the issuing and returning of company equipment to employees.

The sequel introduces the language of EquipmentIssuerProcess.

An employees may be issued company equipment provided that it is in stock and for as long as his manager approves of this. Therefore, it must be possible to register such an approval.

Definition 17: an approval, by a manager, for an employee, allowing the employee to use a specific kind of company equipment *ManagerApproval*

Equipment Issuers must ensure that every employee disposes of the standard issue equipment that goes with the organizational role(s) that (s)he fulfills. If an employee has registered personal equipment, (s)he need not be issued company equipment of such a kind. *CP2.3v1:2.4a*

Agreement 18: Every employee that fulfills an organizational role must be issued all equipment necessary to fulfill this function

Any company equipment other than the standard issue equipment may only be issued to an employee provided that it is in stock and for as long as his manager approves of this *CP2.3v1:2.4a,c*

Agreement 19: Employees that have company equipment that is not standard issue and for which they have no management permission, must return such equipment

The HRM department must keep stock levels to a minimum, yet be able to supply spare equipment to employees if necessary: in case of a malfunction, or when equipment is lost, the employee and HRM department must find a (temporary) solution that provides the employee (temporarily) with sufficient equipment to continue to do his job. Therefore, it must be ensured that there is at least one spare for every kind of equipment. *CP2.3v1:xxx*

Agreement 20: For every kind of equipment that may be issued, there must be at least one spare in stock

2.4 HRMOfficerProcess

This process specifies the responsibilities of HRMOfficers related to the issuing and returning of company equipment to employees.

Employees (except the Director) should be managed, and hence be assigned a manager.

Agreement 21: Every employee, except the Directory, should have a manager

Agreement 22: Directors do not have a manager

Agreement 23: Every employee should be assigned the role 'Employee'

2.5 ManagerProcess

This process specifies the responsibilities of Managers related to the issuing and returning of company equipment to employees.

Management approvals should only be provided by the managers of the employee to which the approval pertains.

Agreement 24: An approval must be issued by the manager of the employee

Company equipment is to be used sparingly which means that employees will be issued no more than one piece of every kind. This implies that managers may not approve equipment kinds that are standard issue.

Agreement 25: Manager approvals should not be given for standard issue equipment

2.6 SecurityOfficerProcess

This process specifies the responsibilities of SecurityOfficers related to the issuing and returning of company equipment to employees.

The sequel introduces the language of SecurityOfficerProcess.

The Security Officer may impose requirements for specific brands and/or types of equipment in order to prevent insecure usage of such equipment. In order to know which requirements pertain to what equipment, and to test whether or not they are met, we must be able to register such requirements.

Definition 26: the specification of a requirement for some equipment types *SecRequirement*

2.7 HRMStatus

Chapter 3

Diagnosis

This chapter provides an analysis of the Ampersand script of ‘CP23’. This analysis is intended for the authors of this script. It can be used to complete the script or to improve possible flaws.

CP23 does not specify which roles may change the contents of which relations.

CP23 assigns rules to roles. The following table shows the rules that are being maintained by a given role.

rule	EquipmentIssuer	HRMOfficer	ExecEngine	Man
Equipment to be issued	×			
Equipment to be taken in	×			
Equipment to be ordered	×			
Assign manager to employee		×		
Directors do not have a manager			×	
Assign employee role to every employee			×	
No manager approvals for standard issue equipment				×
inseqtApprovedProp			×	
deleqtApprovedProp			×	
instypeApprovedProp			×	
deltypeApprovedProp			×	
insneedsToReturnEqt			×	
delneedsToReturnEqt			×	
insallNecessaryEqtHasBeenIssued			×	
delallNecessaryEqtHasBeenIssued			×	
insnoNecessaryEqtHasBeenIssued			×	

delnoNecessaryEqtHasBeenIssued	×
setemplStatusBlack	×
setemplStatusGreen	×
setemplStatusRed	×
setemplStatusYellow	×
setemplStatusGrey	×
setemplStatusBlue	×
setemplStatusOrange	×

Concepts EqtMake, EqtType, EqtSerial, EqtStatus, EqtCompanyID, Yes/No answer, and Status remain without a purpose.

The purpose of relations *eqtMake*, *eqtType*, *eqtSerial*, *eqtKind*, *eqtStatus*, *eqtID*, *maEmployee*, *maManager*, *maEqtKind*, *eqtApprovedProp*, *eqtSecReqt*, *eqtSatReqt*, *eqtApprovedBySecOff*, *typeApprovedProp*, *typeSecReqt*, *typeSatReqt*, *typeApprovedBySecOff*, *needsToReturnEqt*, *allNecessaryEqtHasBeenIssued*, *noNecessaryEqtHasBeenIssued*, and *emplStatus* is not documented.

All concept definitions in this document are used in relations.

Relations *emplName*, *eqtType*, and *eqtStatus* are not used in any rule.

Figure 3.1 shows a conceptual diagram with all relations declared in 'Policy 2.3 vsn 1: Asset Management - Portable Equipment'.

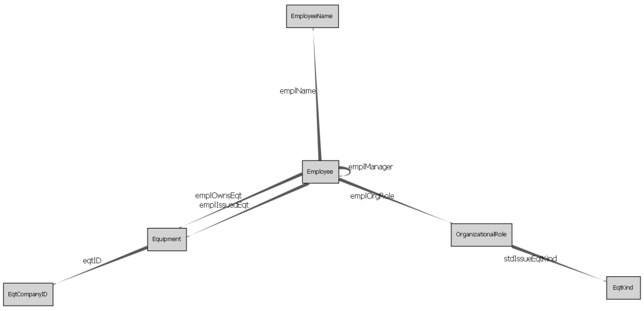

Figure 3.1: Concept diagram of the rules in Policy 2.3 vsn 1: Asset Management - Portable EquipmentDiagnosisConceptualDiagram

Figure 3.2 shows a conceptual diagram with all relations declared in 'Equipment'.

On line numbers 54, 91, 143, and 148 of file .\CP23 Ontology.adl rules are defined without documenting their purpose. On line numbers 201, 204, 217, and 220 of file .\CP23 Ontology.adl and on line numbers 16, 19, 24, 27, 32, 35, 41, 45, 49, 53, 57, 61, and 65 of file .\CP23 Status.adl rules are defined without any explanation.

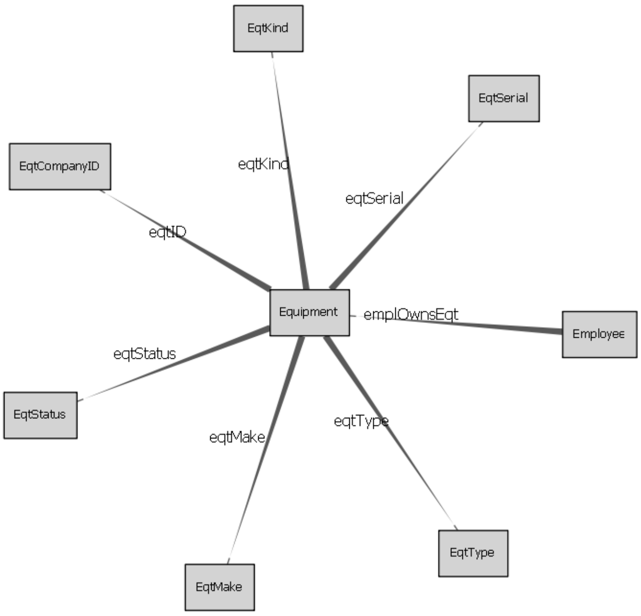

Figure 3.2: Concept diagram of the rules in EquipmentDiagnosisConceptualDiagram

The table below shows for each theme (i.e. process or pattern) the number of relations and rules, followed by the number and percentage that have a reference. Relations declared in multiple themes are counted multiple times.

Theme	Relations	With reference	%	Rules	W
Policy 2.3 vsn 1: Asset Management - Portable Equipment	6	4	66%	3	
Equipment	6	0	0%	2	
EquipmentIssuerProcess	0	0	-	3	
HRMOfficerProcess	0	0	-	3	
ManagerProcess	3	0	0%	2	
SecurityOfficerProcess	8	0	0%	4	
HRMStatus	4	0	0%	13	
Entire context	29	4	13%	30	

The following table shows which rules are not linked to a role within a particular process. This has as consequence that these rule(s) will be maintained by the computer.

process	rule
ManagerProcess	Manager approval integrity, UNI maEmployee::ManagerApproval*Employee, TOT
SecurityOfficerProcess	SYM eqtApprovedProp::Equipment*Equipment, ASY eqtApprovedProp::Equipmen
HRMStatus	UNI emplStatus::Employee*Status

The role-rule assignments in any of the described processes have been assigned to rules within that same process.

Chapter 4

Conceptual Analysis

This chapter defines the formal language, in which functional requirements of 'CP23' can be analysed and expressed.The purpose of this formalisation is to obtain a buildable specification. This chapter allows an independent professional with sufficient background to check whether the agreements made correspond to the formal rules and definitions.

4.1 Policy 2.3 vsn 1: Asset Management - Portable Equipment

Company Inc. has concerns regarding equipment that employees can 'carry around', such as cell-phones, laptops, cars, toolboxes, ID-cards, etc. In order to address these concerns, this policy specifies rules for the purpose of achieving the following objectives: # employees must dispose of all company equipment that is necessary for doing their jobs. # total cost of ownership of company equipment must be controlled, which includes costs for stocks and usage/license fees, in particular when equipment is not or no longer in use. # risks associated with company equipment must be at an acceptable level, not just for company-owned equipment, but also for equipment owned by employees themselves. This pattern defines the agreements necessary to follow the rules that aim to achieve these objectives.

Figure 4.1 shows a conceptual diagram of this pattern.

The definitions of concepts can be found in the glossary.

4.1.1 Declared relations

This section itemizes the declared relations with properties and a meaning.

Within Company Inc. every employee has precisely one name, that identifies the employee. This allows the unambiguous registration of employees.

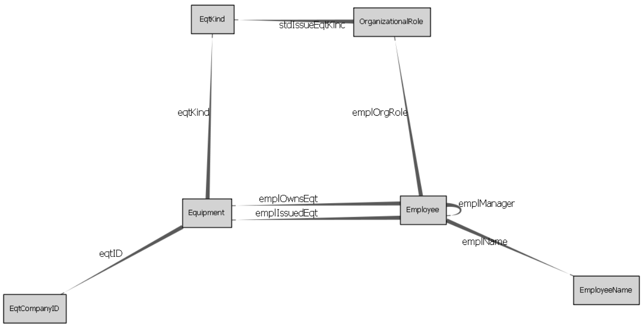

Figure 4.1: Concept diagram of Policy 2.3 vsn 1: Asset Management - Portable Equipment

For this purpose, the following function has been defined

$$emplName \quad : \quad Employee \rightarrow EmployeeName \tag{4.1}$$

Employees have a name

Employees may be issued non-standard equipment provided this is apporved by their manager. Therefore, the manager of employees must be known.

For this purpose, the following irreflexive, antisymmetric, and univalent relation has been defined

$$emplManager \quad : \quad Employee \times Employee \tag{4.2}$$

Employees have been assigned a manager

The responsibilities that employees have are defined by the organizational roles that they fulfill. Depending on such roles, employees will be assigned standard issue equipment. *CP2.3v1:2.4*

For this purpose, the following relation has been defined

$$emplOrgRole \quad : \quad Employee \times OrganizationalRole \tag{4.3}$$

Employees have been assigned (at least) one organizational role that indicate(s) the kind of work they do

Company Inc. has decided to issue equipment to employees based on their organizational role(s). Hence, for every organizational role, it must be possible to define the kinds of equipment that people in such a function must be assigned. This is the so-called the standard issue equipment for the organizational role. *CP2.3v1:3.7, 2.4*

For this purpose, the following relation has been defined

$$stdIssueEqtKind \quad : \quad OrganizationalRole \times EqtKind \tag{4.4}$$

Employees that serve in an organizational role must be assigned equipment of specific kinds

In order to keep good track of portable/mobile equipment that has been issued to employees, every equipment issued to an employee must be registered as such. *CP2.3v1:2.1, 2.4*

For this purpose, the following injective relation has been defined

$$emplIssuedEqt \quad : \quad Employee \times Equipment \tag{4.5}$$

An employee can be issued company equipment (for which it then is responsible)

Employees are allowed to use personal equipment for their work, provided that they register such devices. *CP2.3v1:3.6*

For this purpose, the following injective relation has been defined

$$emplOwnsEqt \quad : \quad Employee \times Equipment \tag{4.6}$$

Employees may use their own portable equipment for their work

The following univalent relation has been defined

$$eqtID \quad : \quad Equipment \times EqtCompanyID \tag{4.7}$$

Company equipment must be identifiable

The following symmetric, antisymmetric, univalent, and injective relation has been defined

$$eqtApprovedProp \quad : \quad Equipment \times Equipment \tag{4.8}$$

Equipment can be approved for use

4.1.2 Formal rules

This section itemizes the formal rules with a reference to the shared language of stakeholders for the sake of traceability.

One employee may only manage another employee if he has been assigned the necessary responsibilities. Such responsibilities are defined for the role 'Manager'. Hence, employees may only be managed by (other) employees that fulfill this role. *CP2.3v1:3.2*

Therefore the following requirement has been defined in section 2.1 p. 6: An employee can only be managed by an employee that fulfills the organizational role of 'Manager'.

This is formalized - using relations 5.11, 5.1 - as

$$emplManager \vdash emplManager; (I_{Employee} \cap emplOrgRole;'\,tManager'; emplOrgRole^{\smile}) \tag{4.9}$$

The following requirement has been defined in section 2.1 p. 6:
Issuing equipment pertains to company equipment only.
This is formalized - using relations 5.3, 4.17 - as

$$I_{Equipment} \cap emplIssuedEqt^{\smile}; emplIssuedEqt \vdash eqtID; eqtID^{\smile} \tag{4.10}$$

All personal equipment and company equipment that is issued to an employee, must be (implicitly or explicitly) approved for use by our Security Officer. *CP2.3v1:2.3-1a*
Therefore the following requirement has been defined in section 2.1 p. 6:
Personal equipment and company equipment that is issued to an employee must have been approved for use.
This is formalized - using relations 5.3, 5.4, 4.8 - as

$$I_{Equipment} \cap (emplIssuedEqt^{\smile}; emplIssuedEqt \cup emplOwnsEqt^{\smile}; emplOwnsEqt) \vdash eqtApprovedProp \tag{4.11}$$

4.2 Equipment

This theme defines the terminology that Company Inc. needs to address concerns related to (portable/mobile) equipment

Figure 4.2 shows a conceptual diagram of this pattern.

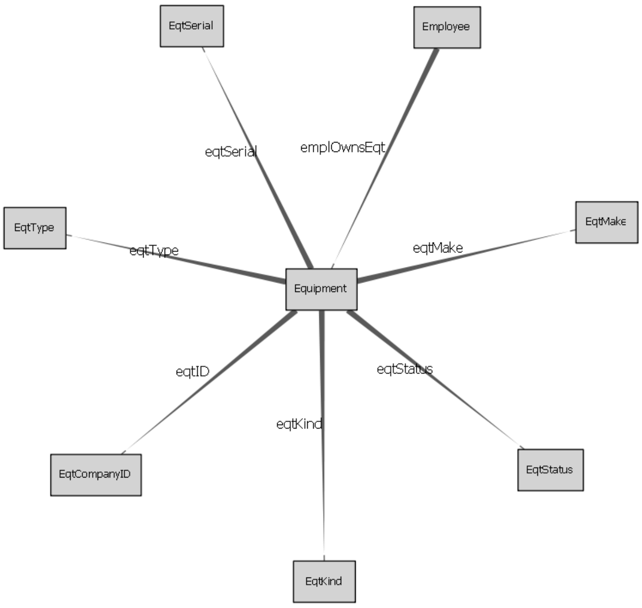

Figure 4.2: Concept diagram of Equipment

The definitions of concepts can be found in the glossary.

4.2.1 Declared relations

This section itemizes the declared relations with properties and a meaning.

The following function has been defined

$$\mathit{eqtMake} \quad : \quad \mathit{Equipment} \rightarrow \mathit{EqtMake} \tag{4.12}$$

Every Equipment has a manufacturer/brand, e.g. 'Dell' or 'Nokia'

The following function has been defined

$$\mathit{eqtType} \quad : \quad \mathit{Equipment} \rightarrow \mathit{EqtType} \tag{4.13}$$

Every Equipment may have one type specified, e.g. 'Inspiron 1234' or 'Passat'

The following function has been defined

$$\mathit{eqtSerial} \quad : \quad \mathit{Equipment} \rightarrow \mathit{EqtSerial} \tag{4.14}$$

Every piece of equipment has a (unique) serial number (manufacturer identifier)

The following function has been defined

$$\mathit{eqtKind} \quad : \quad \mathit{Equipment} \rightarrow \mathit{EqtKind} \tag{4.15}$$

Every Equipment has been assigned (at least) one 'kind', e.g. 'computer', 'cellphone'

The following function has been defined

$$\mathit{eqtStatus} \quad : \quad \mathit{Equipment} \rightarrow \mathit{EqtStatus} \tag{4.16}$$

The functionality status of every Equipment must be known

The following univalent relation has been defined

$$\mathit{eqtID} \quad : \quad \mathit{Equipment} \times \mathit{EqtCompanyID} \tag{4.17}$$

Company equipment must be identifiable

Employees are allowed to use personal equipment for their work, provided that they register such devices. *CP2.3v1:3.6*

For this purpose, the following injective relation has been defined

$$\mathit{emplOwnsEqt} \quad : \quad \mathit{Employee} \times \mathit{Equipment} \tag{4.18}$$

Employees may use their own portable equipment for their work

4.2.2 Formal rules

This section itemizes the formal rules with a reference to the shared language of stakeholders for the sake of traceability.

The following requirement has been defined in section 2.2 p. 6:
Equipment must either have a company ID or be owned by an employee
This is formalized - using relations 4.17, 5.4 - as

$$I_{Equipment} \vdash (eqtID; eqtID^{\smile} \cap \overline{(emplOwnsEqt^{\smile}; emplOwnsEqt)}) \cup (\overline{(eqtID; eqtID^{\smile})} \cap emplOwnsEqt^{\smile}; emp \tag{4.19}$$

In order to do automated reasoning with statuses, and to ensure that statuses remain meaningful, it is necessary to control the allowed values for equipment statuses.
Therefore the following requirement has been defined in section 2.2 p. 6:
Equipment may only be assigned a status 'Functional', 'Not functional' or 'Lost'
This is formalized - using relations - as

$$I_{EqtStatus} =' tFunctional' \cup' tNotfunctional' \cup' tLost' \tag{4.20}$$

Chapter 5

Process Analysis

CP23 does not specify which roles may change the contents of which relations.

CP23 assigns rules to roles. The following table shows the rules that are being maintained by a given role.

Role	Rule
EquipmentIssuer	Equipment to be issued Equipment to be taken in Equipment to be ordered
HRMOfficer	Assign manager to employee
ExecEngine	Directors do not have a manager Assign employee role to every employee inseqtApprovedProp deleqtApprovedProp instypeApprovedProp deltypeApprovedProp insneedsToReturnEqt delneedsToReturnEqt insallNecessaryEqtHasBeenIssued delallNecessaryEqtHasBeenIssued insnoNecessaryEqtHasBeenIssued delnoNecessaryEqtHasBeenIssued setemplStatusBlack setemplStatusGreen setemplStatusRed setemplStatusYellow setemplStatusGrey setemplStatusBlue setemplStatusOrange
Manager	No manager approvals for standard issue equipment

5.1 EquipmentIssuerProcess

This process specifies the responsibilities of an EquipmentIssuer related to the issuing and returning of company equipment to employees.

Figure 5.1 shows the process model.

Figure 5.1: Process model of EquipmentIssuerProcesstxtProcess

The conceptual diagram of figure 5.2 provides an overview of the language in which this process is expressed.

Figure 5.2: Basic sentences of EquipmentIssuerProcessConceptualProcess

Equipment to be issued Equipment Issuers must ensure that every employee disposes of the standard issue equipment that goes with the organizational role(s) that (s)he fulfills. If an employee has registered personal equipment, (s)he need not be issued company equipment of such a kind. *CP2.3v1:2.4a*

The responsibilities that employees have are defined by the organizational roles that they fulfill. Depending on such roles, employees will be assigned standard issue equipment. *CP2.3v1:2.4*

Company Inc. has decided to issue equipment to employees based on their organizational role(s). Hence, for every organizational role, it must be possible to define the kinds of equipment that people in such a function must be assigned. This is the so-called the standard issue equipment for the organizational role. *CP2.3v1:3.7, 2.4*

In order to keep good track of portable/mobile equipment that has been issued to employees, every equipment issued to an employee must be registered as such. *CP2.3v1:2.1, 2.4*

Employees are allowed to use personal equipment for their work, provided that they register such devices. *CP2.3v1:3.6*

To arrive at the formalization in equation 5.6, the following five relations are introduced.

$$emplOrgRole : Employee \times OrganizationalRole \tag{5.1}$$

$$stdIssueEqtKind : OrganizationalRole \times EqtKind \tag{5.2}$$

$$emplIssuedEqt : Employee \times Equipment \tag{5.3}$$

$$emplOwnsEqt : Employee \times Equipment \tag{5.4}$$

$$eqtKind : Equipment \rightarrow EqtKind \tag{5.5}$$

Activities that are defined by this rule are finished when:

$$emplOrgRole; stdIssueEqtKind \vdash (emplIssuedEqt \cup emplOwnsEqt); eqtKind \tag{5.6}$$

This corresponds to 'Equipment to be issued' (2.3 op pg. 7).

Equipment to be taken in Any company equipment other than the standard issue equipment may only be issued to an employee provided that it is in stock and for as long as his manager approves of this *CP2.3v1:2.4a,c*
To arrive at the formalization in equation 5.9, the following two relations are introduced.

$$eqtMake : Equipment \rightarrow EqtMake \tag{5.7}$$

$$eqtSerial : Equipment \rightarrow EqtSerial \tag{5.8}$$

We also use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), **??** (*maEmployee*), and **??** (*maEqtKind*).
Activities that are defined by this rule are finished when:

$$emplIssuedEqt \vdash (emplOrgRole; stdIssueEqtKind \cup maEmployee^{\smile}; maEqtKind); eqtKind^{\smile} \tag{5.9}$$

This corresponds to 'Equipment to be taken in' (2.3 op pg. 7).

Equipment to be ordered The HRM department must keep stock levels to a minimum, yet be able to supply spare equipment to employees if necessary: in case of a malfunction, or when equipment is lost, the employee and HRM department must find a (temporary) solution that provides the employee (temporarily) with sufficient equipment to continue to do his job. Therefore, it must be ensured that there is at least one spare for every kind of equipment. *CP2.3v1:xxx*
We use definitions 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), and 5.5 (*eqtKind*).
Activities that are defined by this rule are finished when:

$$I_{EqtKind} \cap stdIssueEqtKind^{\smile}; stdIssueEqtKind \vdash eqtKind^{\smile}; (I_{Equipment} \cap \overline{(emplIssuedEqt^{\smile}; emplIssued} \tag{5.10}$$

5.2 HRMOfficerProcess

This process specifies the responsibilities of HRMOfficers related to the issuing and returning of company equipment to employees.

Figure 5.3 shows the process model.

Figure 5.3: Process model of HRMOfficerProcesstxtProcess

The conceptual diagram of figure 5.4 provides an overview of the language in which this process is expressed.

Assign manager to employee Employees (except the Director) should be managed, and hence be assigned a manager.

Figure 5.4: Basic sentences of HRMOfficerProcessConceptualProcess

Employees may be issued non-standard equipment provided this is apporved by their manager. Therefore, the manager of employees must be known. In order to formalize this, a relation emplManager is introduced (5.11):

$$emplManager \quad : \quad Employee \times Employee \tag{5.11}$$

Beside that, we use definition 5.1 (emplOrgRole) to formalize requirement 2.4 (page 7):
Activities that are defined by this rule are finished when:

$$I_{Employee} \cap \overline{(emplOrgRole;'tDirector';emplOrgRole^{\smile})} \vdash emplManager;emplManager^{\smile} \tag{5.12}$$

Directors do not have a manager We use definitions 5.11 (*emplManager*) and 5.1 (*emplOrgRole*).
Activities that are defined by this rule are finished when:

$$(I_{Employee} \cap emplOrgRole;'tDirector';emplOrgRole^{\smile});emplManager \vdash \overline{V_{Employeeimes Employee}} \tag{5.13}$$

Assign employee role to every employee We use definition 5.1 (emplOrgRole).
Activities that are defined by this rule are finished when:

$$I_{Employee} \vdash emplOrgRole;'tEmployee';emplOrgRole^{\smile} \tag{5.14}$$

5.3 ManagerProcess

This process specifies the responsibilities of Managers related to the issuing and returning of company equipment to employees.

Figure 5.5 shows the process model.

Figure 5.5: Process model of ManagerProcesstxtProcess

The conceptual diagram of figure 5.6 provides an overview of the language in which this process is expressed.

Figure 5.6: Basic sentences of ManagerProcessConceptualProcess

Manager approval integrity Management approvals should only be provided by the managers of the employee to which the approval pertains.
We use definitions 5.11 (*emplManager*), **??** (*maEmployee*), and **??** (*maManager*).
This means:

$$maEmployee^{\smile}; maManager \vdash emplManager \tag{5.15}$$

No manager approvals for standard issue equipment Company equipment is to be used sparingly which means that employees will be issued no more than one piece of every kind. This implies that managers may not approve equipment kinds that are standard issue.
We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), **??** (*maEmployee*), and **??** (*maEqtKind*).
Activities that are defined by this rule are finished when:

$$emplOrgRole; stdIssueEqtKind \vdash \overline{(maEmployee^{\smile}; maEqtKind)} \tag{5.16}$$

5.4 SecurityOfficerProcess

This process specifies the responsibilities of SecurityOfficers related to the issuing and returning of company equipment to employees.

Figure 5.7 shows the process model.

Figure 5.7: Process model of SecurityOfficerProcesstxtProcess

The conceptual diagram of figure 5.8 provides an overview of the language in which this process is expressed.

Figure 5.8: Basic sentences of SecurityOfficerProcessConceptualProcess

inseqtApprovedProp We use definitions 4.8 (*eqtApprovedProp*), **??** (*eqtSecReqt*), **??** (*eqtSatReqt*), and **??** (*eqtApprovedBySecOff*).
Activities that are defined by this rule are finished when:

$$I_{Equipment} \cap (eqtApprovedBySecOff;' tYes'; eqtApprovedBySecOff^{\smile} \cup \overline{eqtSecReqt} \dagger eqtSatReqt^{\smile}) \vdash eqtApp \tag{5.17}$$

deleqtApprovedProp We use definitions 4.8 (*eqtApprovedProp*), **??** (*eqtSecReqt*), **??** (*eqtSatReqt*), and **??** (*eqtApprovedBySecOff*).
Activities that are defined by this rule are finished when:

$$eqtApprovedProp \vdash I_{Equipment} \cap (eqtApprovedBySecOff;' tYes'; eqtApprovedBySecOff^{\smile} \cup \overline{eqtSecReqt} \dagger eqt \tag{5.18}$$

instypeApprovedProp We use definitions **??** (*typeApprovedProp*), **??** (*typeSecReqt*), **??** (*typeSatReqt*), and **??** (*typeApprovedBySecOff*).
Activities that are defined by this rule are finished when:

$$I_{EqtType} \cap (typeApprovedBySecOff; 'tYes'; typeApprovedBySecOff^{\smile} \cup \overline{typeSecReqt} \dagger typeSatReqt^{\smile}) \vdash type \tag{5.19}$$

deltypeApprovedProp We use definitions **??** (*typeApprovedProp*), **??** (*typeSecReqt*), **??** (*typeSatReqt*), and **??** (*typeApprovedBySecOff*).
Activities that are defined by this rule are finished when:

$$typeApprovedProp \vdash I_{EqtType} \cap (typeApprovedBySecOff; 'tYes'; typeApprovedBySecOff^{\smile} \cup \overline{typeSecReqt} \dagger t \tag{5.20}$$

5.5 HRMStatus

Figure 5.9 shows the process model.

Figure 5.9: Process model of HRMStatustxtProcess

The conceptual diagram of figure 5.10 provides an overview of the language in which this process is expressed.

Figure 5.10: Basic sentences of HRMStatusConceptualProcess

insneedsToReturnEqt We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), and **??** (*needsToReturnEqt*).
Activities that are defined by this rule are finished when:

$$I_{Employee} \cap (emplIssuedEqt; eqtKind \cap \overline{(emplOrgRole; stdIssueEqtKind)}); V_{EqtKind_{imes}Employee} \vdash needs \tag{5.21}$$

delneedsToReturnEqt We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), and **??** (*needsToReturnEqt*).
Activities that are defined by this rule are finished when:

$$needsToReturnEqt \vdash I_{Employee} \cap (emplIssuedEqt; eqtKind \cap \overline{(emplOrgRole; stdIssueEqtKind)}); V_{EqtKind} \tag{5.22}$$

insallNecessaryEqtHasBeenIssued We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), and **??** (*allNecessaryEqtHasBeenIssued*).
Activities that are defined by this rule are finished when:

$$I_{Employee}\cap\overline{(\mathit{emplOrgRole};\mathit{stdIssueEqtKind})}\dagger(\mathit{emplIssuedEqt};\mathit{eqtKind})^{\smile}\vdash \mathit{allNecessaryEqtHasBeenIssu} \tag{5.23}$$

delallNecessaryEqtHasBeenIssued We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), and **??** (*allNecessaryEqtHasBeenIssued*).
Activities that are defined by this rule are finished when:

$$\mathit{allNecessaryEqtHasBeenIssued}\vdash I_{Employee}\cap\overline{(\mathit{emplOrgRole};\mathit{stdIssueEqtKind})}\dagger(\mathit{emplIssuedEqt};\mathit{eqtKind} \tag{5.24}$$

insnoNecessaryEqtHasBeenIssued We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), and **??** (*noNecessaryEqtHasBeenIssued*).
Activities that are defined by this rule are finished when:

$$I_{Employee}\cap\overline{(\mathit{emplOrgRole};\mathit{stdIssueEqtKind})}\dagger\overline{(\mathit{emplIssuedEqt};\mathit{eqtKind})}^{\smile}\vdash \mathit{noNecessaryEqtHasBeenIssu} \tag{5.25}$$

delnoNecessaryEqtHasBeenIssued We use definitions 5.1 (*emplOrgRole*), 5.2 (*stdIssueEqtKind*), 5.3 (*emplIssuedEqt*), 5.5 (*eqtKind*), and **??** (*noNecessaryEqtHasBeenIssued*).
Activities that are defined by this rule are finished when:

$$\mathit{noNecessaryEqtHasBeenIssued}\vdash I_{Employee}\cap\overline{(\mathit{emplOrgRole};\mathit{stdIssueEqtKind})}\dagger\overline{(\mathit{emplIssuedEqt};\mathit{eqtKind}} \tag{5.26}$$

setemplStatusBlack We use definitions **??** (*needsToReturnEqt*), **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).
Activities that are defined by this rule are finished when:

$$I_{Employee}\cap\overline{\mathit{needsToReturnEqt}}\cap\mathit{allNecessaryEqtHasBeenIssued}\cap\mathit{noNecessaryEqtHasBeenIssued}\vdash \mathit{empl} \tag{5.27}$$

setemplStatusGreen We use definitions **??** (*needsToReturnEqt*), **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).
Activities that are defined by this rule are finished when:

$$I_{Employee}\cap\overline{\mathit{needsToReturnEqt}}\cap\mathit{allNecessaryEqtHasBeenIssued}\cap\overline{\mathit{noNecessaryEqtHasBeenIssued}}\vdash \mathit{empl} \tag{5.28}$$

setemplStatusRed We use definitions **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).

Activities that are defined by this rule are finished when:

$$I_{Employee} \cap \overline{\mathit{allNecessaryEqtHasBeenIssued}} \cap \mathit{noNecessaryEqtHasBeenIssued} \vdash \mathit{emplStatus};'\mathit{tRed}';\mathit{emplS} \tag{5.29}$$

setemplStatusYellow We use definitions **??** (*needsToReturnEqt*), **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).

Activities that are defined by this rule are finished when:

$$I_{Employee} \cap \overline{\mathit{needsToReturnEqt}} \cap \overline{\mathit{allNecessaryEqtHasBeenIssued}} \cap \overline{\mathit{noNecessaryEqtHasBeenIssued}} \vdash \mathit{empl} \tag{5.30}$$

setemplStatusGrey We use definitions **??** (*needsToReturnEqt*), **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).

Activities that are defined by this rule are finished when:

$$I_{Employee} \cap \mathit{needsToReturnEqt} \cap \mathit{allNecessaryEqtHasBeenIssued} \cap \mathit{noNecessaryEqtHasBeenIssued} \vdash \mathit{empl} \tag{5.31}$$

setemplStatusBlue We use definitions **??** (*needsToReturnEqt*), **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).

Activities that are defined by this rule are finished when:

$$I_{Employee} \cap \mathit{needsToReturnEqt} \cap \mathit{allNecessaryEqtHasBeenIssued} \cap \overline{\mathit{noNecessaryEqtHasBeenIssued}} \vdash \mathit{empl} \tag{5.32}$$

setemplStatusOrange We use definitions **??** (*needsToReturnEqt*), **??** (*allNecessaryEqtHasBeenIssued*), **??** (*noNecessaryEqtHasBeenIssued*), and **??** (*emplStatus*).

Activities that are defined by this rule are finished when:

$$I_{Employee} \cap \mathit{needsToReturnEqt} \cap \overline{\mathit{allNecessaryEqtHasBeenIssued}} \cap \overline{\mathit{noNecessaryEqtHasBeenIssued}} \vdash \mathit{empl} \tag{5.33}$$

Chapter 6

Data structure

This chapter contains the result of the data analisys. It is structured as follows:

We start with the classification model, followed by a list of all relations, that are the foundation of the rest of the analisys. Finally, the logical and technical data model are discussed.

6.1 Classifications

No classifications have been defined

6.2 Fact types

This section enumerates the fact types, that have been used in the design of the datastructure. For each fact type its name, the source and target concept and the properties are documented.

emplName : ***Employee*** × ***EmployeeName*** Employees have a name

Properties: UNI, TOT

emplManager : ***Employee*** × ***Employee*** Employees have been assigned a manager

Properties: IRF, ASY, UNI, TOT, SUR

emplOrgRole : ***Employee*** × ***OrganizationalRole*** Employees have been assigned (at least) one organizational role that indicate(s) the kind of work they do

Properties: --

stdIssueEqtKind : ***OrganizationalRole*** × ***EqtKind*** Employees that serve in an organizational role must be assigned equipment of specific kinds

Properties: --

emplIssuedEqt : ***Employee*** × ***Equipment*** An employee can be issued company equipment (for which it then is responsible)

Properties: INJ

emplOwnsEqt : ***Employee*** × ***Equipment*** Employees may use their own portable equipment for their work

Properties: INJ, SUR

eqtMake : ***Equipment*** × ***EqtMake*** Every Equipment has a manufacturer/brand, e.g. 'Dell' or 'Nokia'

Properties: UNI, TOT

eqtType : ***Equipment*** × ***EqtType*** Every Equipment may have one type specified, e.g. 'Inspiron 1234' or 'Passat'

Properties: UNI, TOT

eqtSerial : ***Equipment*** × ***EqtSerial*** Every piece of equipment has a (unique) serial number (manufacturer identifier)

Properties: UNI, TOT

eqtKind : ***Equipment*** × ***EqtKind*** Every Equipment has been assigned (at least) one 'kind', e.g. 'computer', 'cellphone'

Properties: UNI, TOT

eqtStatus : ***Equipment*** × ***EqtStatus*** The functionality status of every Equipment must be known

Properties: UNI, TOT

eqtID : ***Equipment*** × ***EqtCompanyID*** Company equipment must be identifiable

Properties: UNI, TOT

maEmployee : ***ManagerApproval*** × ***Employee*** **Properties**: UNI, TOT

maManager : ***ManagerApproval*** × ***Employee*** **Properties**: UNI, TOT

maEqtKind : ***ManagerApproval*** × ***EqtKind*** **Properties**: --

eqtApprovedProp : ***Equipment*** × ***Equipment*** Equipment can be approved for use

Properties: SYM, ASY, UNI, INJ

eqtSecReqt : ***Equipment*** × ***SecRequirement*** For specific equipment, security requirements may need to be satisfied

Properties: --

eqtSatReqt : ***Equipment*** × ***SecRequirement*** Equipment may satisfy security requirements

Properties: --

eqtApprovedBySecOff : ***Equipment*** × ***Yes/No answer*** Equipment can manually be approved by the Security Officer

Properties: --

typeApprovedProp : ***EqtType*** × ***EqtType*** Equipment types can be approved for use

Properties: SYM, ASY, UNI, INJ

typeSecReqt : ***EqtType*** × ***SecRequirement*** For specific equipement types, security requirements may need to be satisfied

Properties: --

typeSatReqt : ***EqtType*** × ***SecRequirement*** For all equipment of some type, security requirements may be satisfied

Properties: --

typeApprovedBySecOff : ***EqtType*** × ***Yes/No answer*** Equipment types can manually be approved by the Security Officer

Properties: --

needsToReturnEqt : ***Employee*** × ***Employee*** **Properties**: --

allNecessaryEqtHasBeenIssued : ***Employee*** × ***Employee*** **Properties**: --

noNecessaryEqtHasBeenIssued : ***Employee*** × ***Employee*** **Properties**: --

emplStatus : ***Employee*** × ***Status*** **Properties**: UNI

6.3 Logical datamodel

The functional requirements have been translated into a data model. This model is shown by figure 6.1.

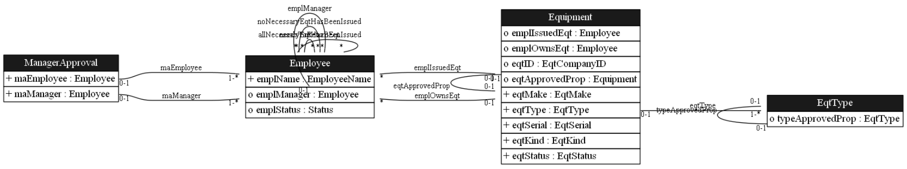

Figure 6.1: Logical data model of CP23

There are four entity types.The details of each entity type are described (in alfabetical order) in the following paragraphs:

6.3.1 Entity type: ***Employee***

This entity type has the following attributes:

Attribute	Type	
Id	Employee	Primary key
emplName	EmployeeName	Mandatory
emplManager	Employee	Optional
emplStatus	Status	Optional

Employee has the following associations:

1. Every *Employee* ‘emplManager’ zero or more *Employee.* For the other way round, for this relation holds that each *Employee* at most one *Employee.*
2. Every *Employee* ‘emplIssuedEqt’ at most one *Equipment.* For the other way round, for this relation holds that each *Equipment* zero or more *Employee.*
3. Every *Employee* ‘emplOwnsEqt’ at most one *Equipment.* For the other way round, for this relation holds that each *Equipment* zero or more *Employee.*
4. Every *ManagerApproval* must ‘maEmployee’ at least one *Employee.* For the other way round, for this relation holds that each *Employee* at most one *ManagerApproval.*
5. Every *ManagerApproval* must ‘maManager’ at least one *Employee.* For the other way round, for this relation holds that each *Employee* at most one *ManagerApproval.*
6. Every *Employee* ‘needsToReturnEqt’ zero or more *Employee.* For the other way round, for this relation holds that each *Employee* zero or more *Employee.*
7. Every *Employee* ‘allNecessaryEqtHasBeenIssued’ zero or more *Employee.* For the other way round, for this relation holds that each *Employee* zero or more *Employee.*
8. Every *Employee* ‘noNecessaryEqtHasBeenIssued’ zero or more *Employee.* For the other way round, for this relation holds that each *Employee* zero or more *Employee.*

6.3.2 Entity type: ***EqtType***

This entity type has the following attributes:

Attribute	Type	
Id	EqtType	Primary key
typeApprovedProp	EqtType	Optional

EqtType has the following associations:

1. Every *Equipment* must ‘eqtType’ at least one *EqtType*. For the other way round, for this relation holds that each *EqtType* at most one *Equipment*.
2. Every *EqtType* ‘typeApprovedProp’ at most one *EqtType*. For the other way round, for this relation holds that each *EqtType* at most one *EqtType*.

6.3.3 Entity type: *Equipment*

This entity type has the following attributes:

Attribute	Type	
Id	Equipment	Primary key
emplIssuedEqt	Employee	Optional
emplOwnsEqt	Employee	Optional
eqtID	EqtCompanyID	Optional
eqtApprovedProp	Equipment	Optional
eqtMake	EqtMake	Mandatory
eqtType	EqtType	Mandatory
eqtSerial	EqtSerial	Mandatory
eqtKind	EqtKind	Mandatory
eqtStatus	EqtStatus	Mandatory

Equipment has the following associations:

1. Every *Employee* ‘emplIssuedEqt’ at most one *Equipment*. For the other way round, for this relation holds that each *Equipment* zero or more *Employee*.
2. Every *Employee* ‘emplOwnsEqt’ at most one *Equipment*. For the other way round, for this relation holds that each *Equipment* zero or more *Employee*.
3. Every *Equipment* ‘eqtApprovedProp’ at most one *Equipment*. For the other way round, for this relation holds that each *Equipment* at most one *Equipment*.
4. Every *Equipment* must ‘eqtType’ at least one *EqtType*. For the other way round, for this relation holds that each *EqtType* at most one *Equipment*.

6.3.4 Entity type: *ManagerApproval*

This entity type has the following attributes:

Attribute	Type	
Id	ManagerApproval	Primary key
maEmployee	Employee	Mandatory
maManager	Employee	Mandatory

ManagerApproval has the following associations:

1. Every *ManagerApproval* must 'maEmployee' at least one *Employee*. For the other way round, for this relation holds that each *Employee* at most one *ManagerApproval*.
2. Every *ManagerApproval* must 'maManager' at least one *Employee*. For the other way round, for this relation holds that each *Employee* at most one *ManagerApproval*.

6.4 Technical datamodel

The functional requirements have been translated into a technical data model. This model is shown by figure 6.2.

The technical datamodel consists of the following 28tables:

6.4.1 Table: Employee

This table has the following 4 fields:

- **Employee**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.
- **emplName**
  This attribute implements the relation $Employee \xrightarrow{emplName} EmployeeName$.
  `SQLVarchar 255`, Optional.
- **emplManager**
  This attribute implements the relation $Employee \xrightarrow{emplManager} Employee$.
  `SQLVarchar 255`, Optional.
- **emplStatus**
  This attribute implements the relation $Employee \xrightarrow{emplStatus} Status$.
  `SQLVarchar 255`, Optional.

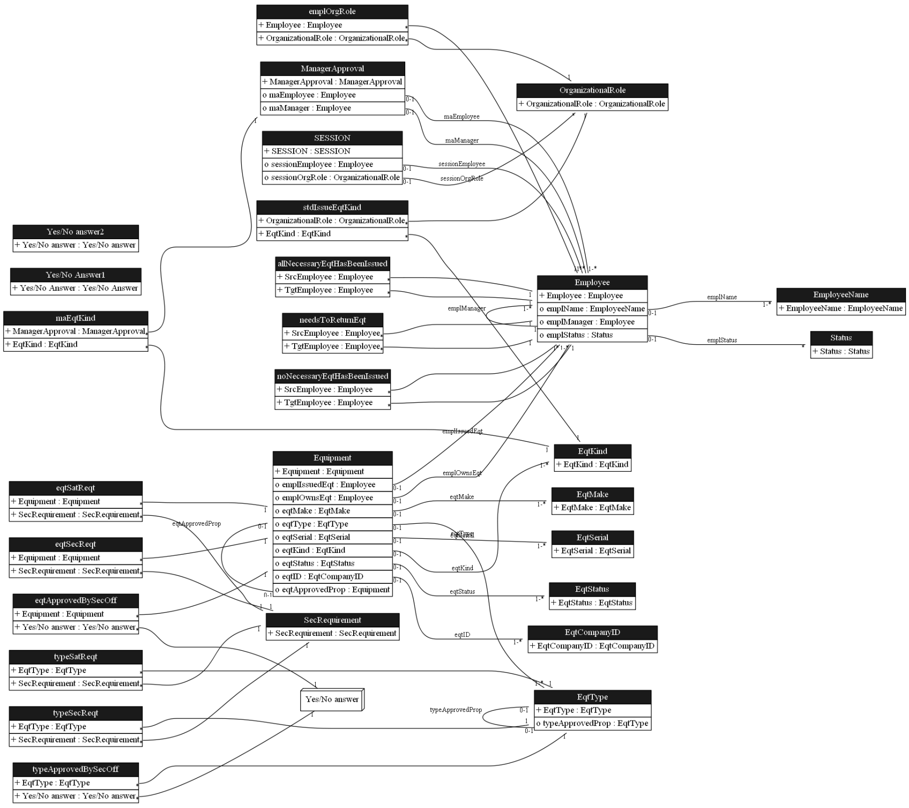

Figure 6.2: Technical data model of CP23

6.4.2 Table: EmployeeName

This table has the following 1 fields:

- **EmployeeName**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.3 Table: EqtCompanyID

This table has the following 1 fields:

- **EqtCompanyID**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.4 Table: EqtKind

This table has the following 1 fields:

- **EqtKind**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.5 Table: EqtMake

This table has the following 1 fields:

- **EqtMake**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.6 Table: EqtSerial

This table has the following 1 fields:

- **EqtSerial**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.7 Table: EqtStatus

This table has the following 1 fields:

- **EqtStatus**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.8 Table: EqtType

This table has the following 2 fields:

- **EqtType**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.
- **typeApprovedProp**
  This attribute implements the relation $EqtType \xrightarrow{typeApprovedProp} EqtType$.
  `SQLVarchar 255`, Optional, Unique.

6.4.9 Table: Equipment

This table has the following 10 fields:

- **Equipment**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.
- **emplIssuedEqt**
  This attribute implements the relation $Employee \xleftarrow{emplIssuedEqt} Equipment$.
  `SQLVarchar 255`, Optional.
- **emplOwnsEqt**
  This attribute implements the relation $Employee \xleftarrow{emplOwnsEqt} Equipment$.
  `SQLVarchar 255`, Optional.
- **eqtMake**
  This attribute implements the relation $Equipment \xrightarrow{eqtMake} EqtMake$.
  `SQLVarchar 255`, Optional.
- **eqtType**
  This attribute implements the relation $Equipment \xrightarrow{eqtType} EqtType$.
  `SQLVarchar 255`, Optional.
- **eqtSerial**
  This attribute implements the relation $Equipment \xrightarrow{eqtSerial} EqtSerial$.
  `SQLVarchar 255`, Optional.
- **eqtKind**
  This attribute implements the relation $Equipment \xrightarrow{eqtKind} EqtKind$.
  `SQLVarchar 255`, Optional.
- **eqtStatus**
  This attribute implements the relation $Equipment \xrightarrow{eqtStatus} EqtStatus$.
  `SQLVarchar 255`, Optional.
- **eqtID**
  This attribute implements the relation $Equipment \xrightarrow{eqtID} EqtCompanyID$.
  `SQLVarchar 255`, Optional.
- **eqtApprovedProp**
  This attribute implements the relation $Equipment \xrightarrow{eqtApprovedProp} Equipment$.
  `SQLVarchar 255`, Optional, Unique.

6.4.10 Table: ManagerApproval

This table has the following 3 fields:

- **ManagerApproval**
This attribute is the primary key.
`SQLVarchar 255`, Mandatory, Unique.

- **maEmployee**
This attribute implements the relation $ManagerApproval \xrightarrow{maEmployee} Employee.$
`SQLVarchar 255`, Optional.

- **maManager**
This attribute implements the relation $ManagerApproval \xrightarrow{maManager} Employee.$
`SQLVarchar 255`, Optional.

6.4.11 Table: OrganizationalRole

This table has the following 1 fields:

- **OrganizationalRole**
This attribute is the primary key.
`SQLVarchar 255`, Mandatory, Unique.

6.4.12 Table: SESSION

This table has the following 3 fields:

- **SESSION**
This attribute is the primary key.
`SQLVarchar 255`, Mandatory, Unique.

- **sessionEmployee**
This attribute implements the relation $SESSION \xrightarrow{sessionEmployee} Employee.$
`SQLVarchar 255`, Optional.

- **sessionOrgRole**
This attribute implements the relation $SESSION \xrightarrow{sessionOrgRole} OrganizationalRole.$
`SQLVarchar 255`, Optional.

6.4.13 Table: SecRequirement

This table has the following 1 fields:

- **SecRequirement**
This attribute is the primary key.
`SQLBlob`, Mandatory, Unique.

6.4.14 Table: Status

This table has the following 1 fields:

- **Status**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.15 Table: Yes/No Answer1

This table has the following 1 fields:

- **Yes/No Answer**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.16 Table: Yes/No answer2

This table has the following 1 fields:

- **Yes/No answer**
  This attribute is the primary key.
  `SQLVarchar 255`, Mandatory, Unique.

6.4.17 Table: allNecessaryEqtHasBeenIssued

This is a link-table, implementing the relation $Employee \xrightarrow{allNecessaryEqtHasBeenIssued} Employee$. It contains the following columns:

- **SrcEmployee**
  This attribute is a foreign key to Employee
  `SQLVarchar 255`, Mandatory.
- **TgtEmployee**
  This attribute implements the relation $Employee \xrightarrow{allNecessaryEqtHasBeenIssued} Employee$.
  `SQLVarchar 255`, Mandatory.

6.4.18 Table: emplOrgRole

This is a link-table, implementing the relation $Employee \xrightarrow{emplOrgRole} OrganizationalRole$. It contains the following columns:

- **Employee**
  This attribute is a foreign key to Employee
  `SQLVarchar 255`, Mandatory.

- **OrganizationalRole**
  This attribute implements the relation $Employee \xrightarrow{emplOrgRole} OrganizationalRole$.
  `SQLVarchar 255`, Mandatory.

6.4.19 Table: eqtApprovedBySecOff

This is a link-table, implementing the relation $Equipment \xrightarrow{eqtApprovedBySecOff} Yes/No answer$. It contains the following columns:

- **Equipment**
  This attribute is a foreign key to Equipment
  `SQLVarchar 255`, Mandatory.
- **Yes/No answer**
  This attribute implements the relation $Equipment \xrightarrow{eqtApprovedBySecOff} Yes/No answer$.
  `SQLVarchar 255`, Mandatory.

6.4.20 Table: eqtSatReqt

This is a link-table, implementing the relation $Equipment \xrightarrow{eqtSatReqt} SecRequirement$. It contains the following columns:

- **Equipment**
  This attribute is a foreign key to Equipment
  `SQLVarchar 255`, Mandatory.
- **SecRequirement**
  This attribute implements the relation $Equipment \xrightarrow{eqtSatReqt} SecRequirement$.
  `SQLBlob`, Mandatory.

6.4.21 Table: eqtSecReqt

This is a link-table, implementing the relation $Equipment \xrightarrow{eqtSecReqt} SecRequirement$. It contains the following columns:

- **Equipment**
  This attribute is a foreign key to Equipment
  `SQLVarchar 255`, Mandatory.
- **SecRequirement**
  This attribute implements the relation $Equipment \xrightarrow{eqtSecReqt} SecRequirement$.
  `SQLBlob`, Mandatory.

6.4.22 Table: maEqtKind

This is a link-table, implementing the relation $ManagerApproval \xrightarrow{maEqtKind} EqtKind$. It contains the following columns:

- **ManagerApproval**
  This attribute is a foreign key to ManagerApproval
  `SQLVarchar 255`, Mandatory.
- **EqtKind**
  This attribute implements the relation $ManagerApproval \xrightarrow{maEqtKind} EqtKind$.
  `SQLVarchar 255`, Mandatory.

6.4.23 Table: needsToReturnEqt

This is a link-table, implementing the relation $Employee \xrightarrow{needsToReturnEqt} Employee$. It contains the following columns:

- **SrcEmployee**
  This attribute is a foreign key to Employee
  `SQLVarchar 255`, Mandatory.
- **TgtEmployee**
  This attribute implements the relation $Employee \xrightarrow{needsToReturnEqt} Employee$.
  `SQLVarchar 255`, Mandatory.

6.4.24 Table: noNecessaryEqtHasBeenIssued

This is a link-table, implementing the relation $Employee \xrightarrow{noNecessaryEqtHasBeenIssued} Employee$. It contains the following columns:

- **SrcEmployee**
  This attribute is a foreign key to Employee
  `SQLVarchar 255`, Mandatory.
- **TgtEmployee**
  This attribute implements the relation $Employee \xrightarrow{noNecessaryEqtHasBeenIssued} Employee$.
  `SQLVarchar 255`, Mandatory.

6.4.25 Table: stdIssueEqtKind

This is a link-table, implementing the relation $OrganizationalRole \xrightarrow{stdIssueEqtKind} EqtKind$. It contains the following columns:

- **OrganizationalRole**
  This attribute is a foreign key to OrganizationalRole
  `SQLVarchar 255`, Mandatory.
- **EqtKind**
  This attribute implements the relation $OrganizationalRole \xrightarrow{stdIssueEqtKind} EqtKind$.
  `SQLVarchar 255`, Mandatory.

6.4.26 Table: typeApprovedBySecOff

This is a link-table, implementing the relation $EqtType \xrightarrow{typeApprovedBySecOff} Yes/Noanswer$. It contains the following columns:

- **EqtType**
  This attribute is a foreign key to EqtType
  `SQLVarchar 255`, Mandatory.
- **Yes/No answer**
  This attribute implements the relation $EqtType \xrightarrow{typeApprovedBySecOff} Yes/Noanswer$.
  `SQLVarchar 255`, Mandatory.

6.4.27 Table: typeSatReqt

This is a link-table, implementing the relation $EqtType \xrightarrow{typeSatReqt} SecRequirement$. It contains the following columns:

- **EqtType**
  This attribute is a foreign key to EqtType
  `SQLVarchar 255`, Mandatory.
- **SecRequirement**
  This attribute implements the relation $EqtType \xrightarrow{typeSatReqt} SecRequirement$.
  `SQLBlob`, Mandatory.

6.4.28 Table: typeSecReqt

This is a link-table, implementing the relation $EqtType \xrightarrow{typeSecReqt} SecRequirement$. It contains the following columns:

- **EqtType**
  This attribute is a foreign key to EqtType
  `SQLVarchar 255`, Mandatory.
- **SecRequirement**
  This attribute implements the relation $EqtType \xrightarrow{typeSecReqt} SecRequirement$.
  `SQLBlob`, Mandatory.

Chapter 7

ECA rules (Flash points)

This chapter lists the ECA rules.

ECA rules:
temporarily not documented

```
ON INSERT Delta IN emplName[Employee*EmployeeName] EXECUTE    -- (ECA rule 1)
ONE OF INSERT INTO Isn{detyp=EmployeeName}
        SELECTFROM ((emplName \/ Delta)~;emplName /\ -I[EmployeeName]) \/ ((empl

       (TO MAINTAIN  -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName:
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (Delta;Delta~ /\ I[Employee]) - I[Employee]

       INSERT INTO Isn{detyp=EmployeeName}
        SELECTFROM (Delta~;Delta /\ I[EmployeeName]) - I[EmployeeName]

(MAINTAINING -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Employe
(MAINTAINING -I[Employee] \/ emplName;emplName~ FROM TOT emplName::Employee*Empl
```

------ Derivation ----->

```
ONE OF INSERT INTO Isn{detyp=EmployeeName}
        SELECTFROM ((emplName \/ Delta)~;emplName /\ -I[EmployeeName]) \/ ((emplName

       (TO MAINTAIN  -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Empl
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (Delta;Delta~ /\ I[Employee]) - I[Employee]

       INSERT INTO Isn{detyp=EmployeeName}
        SELECTFROM (Delta~;Delta /\ I[EmployeeName]) - I[EmployeeName]
```

```
(MAINTAINING -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Employee*Emp
(MAINTAINING -I[Employee] \/ emplName;emplName~ FROM TOT emplName::Employee*EmployeeN
```

<------End Derivation --

```
ON DELETE Delta FROM emplName[Employee*EmployeeName] EXECUTE    -- (ECA rule 2)
DELETE FROM Isn{detyp=Employee}
 SELECTFROM -((emplName /\ -Delta);(emplName /\ -Delta)~) /\ I[Employee]

(TO MAINTAIN  -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Employ
(TO MAINTAIN  -I[Employee] \/ emplName;emplName~ FROM TOT emplName::Employee*Empl
```

------ Derivation ----->

```
DELETE FROM Isn{detyp=Employee}
 SELECTFROM -((emplName /\ -Delta);(emplName /\ -Delta)~) /\ I[Employee]

(TO MAINTAIN  -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Employee*Em
(TO MAINTAIN  -I[Employee] \/ emplName;emplName~ FROM TOT emplName::Employee*Employee
```

<------End Derivation --

```
ON INSERT Delta IN emplManager[Employee*Employee] EXECUTE    -- (ECA rule 3)
BLOCK
(CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
(CANNOT CHANGE V[Employee*Employee] FROM IRF emplManager::Employee*Employee)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
(CANNOT CHANGE V[Employee*Employee] FROM IRF emplManager::Employee*Employee)
```

<------End Derivation --

```
ON DELETE Delta FROM emplManager[Employee*Employee] EXECUTE    -- (ECA rule 4)
ONE OF DELETE FROM emplManager[Employee*Employee]
        SELECTFROM -((emplManager /\ -Delta);(I[Employee] /\ emplOrgRole;'Manage

       (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Ma
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -((emplManager /\ -Delta);(emplManager /\ -Delta)~) /\ -(empl
       (TO MAINTAIN  -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Di
       DELETE FROM maEmployee[ManagerApproval*Employee]
        SELECTFROM maManager;((-emplManager~ /\ maManager~;maEmployee) \/ (Delta

       (TO MAINTAIN  -(maEmployee~;maManager) \/ emplManager FROM Manager approv
       DELETE FROM maManager[ManagerApproval*Employee]
        SELECTFROM maEmployee;((-emplManager /\ maEmployee~;maManager) \/ (Delta

       (TO MAINTAIN  -(maEmployee~;maManager) \/ emplManager FROM Manager approv
       DELETE FROM maManager[ManagerApproval*Employee]
        SELECTFROM -(maEmployee;(emplManager /\ -Delta)) /\ maManager

       (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager approval
       DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -(maEmployee;(emplManager /\ -Delta);maManager~) /\ I[Manager

       (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FR
       DELETE FROM maEmployee[ManagerApproval*Employee]
        SELECTFROM -(maManager;(emplManager~ /\ -Delta~)) /\ maEmployee

       (TO MAINTAIN  -maEmployee~ \/ emplManager;maManager~ FROM Manager approva
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
(MAINTAINING -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Director'[
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
```

------ Derivation ----->

```
ONE OF DELETE FROM emplManager[Employee*Employee]
        SELECTFROM -((emplManager /\ -Delta);(I[Employee] /\ emplOrgRole;'Manager'[O

       (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -((emplManager /\ -Delta);(emplManager /\ -Delta)~) /\ -(emplOrgRo

       (TO MAINTAIN  -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Directo
       DELETE FROM maEmployee[ManagerApproval*Employee]
        SELECTFROM maManager;((-emplManager~ /\ maManager~;maEmployee) \/ (Delta~ /\

       (TO MAINTAIN  -(maEmployee~;maManager) \/ emplManager FROM Manager approval in
       DELETE FROM maManager[ManagerApproval*Employee]
        SELECTFROM maEmployee;((-emplManager /\ maEmployee~;maManager) \/ (Delta /\ m

       (TO MAINTAIN  -(maEmployee~;maManager) \/ emplManager FROM Manager approval in
       DELETE FROM maManager[ManagerApproval*Employee]
        SELECTFROM -(maEmployee;(emplManager /\ -Delta)) /\ maManager

       (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager approval integ
       DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -(maEmployee;(emplManager /\ -Delta);maManager~) /\ I[ManagerAppro

       (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM Ma
       DELETE FROM maEmployee[ManagerApproval*Employee]
        SELECTFROM -(maManager;(emplManager~ /\ -Delta~)) /\ maEmployee

       (TO MAINTAIN  -maEmployee~ \/ emplManager;maManager~ FROM Manager approval int
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
(MAINTAINING -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Director'[Organ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
```

<------End Derivation --

```
          ON INSERT Delta IN emplOrgRole[Employee*OrganizationalRole] EXECUTE    -- (ECA r
          BLOCK
          (CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
          (CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue e
```

------ Derivation ----->

```
     BLOCK
     (CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
     (CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue equip
```

<------End Derivation --

```
          ON DELETE Delta FROM emplOrgRole[Employee*OrganizationalRole] EXECUTE    -- (ECA
          ALL of DELETE FROM emplManager[Employee*Employee]
                  SELECTFROM -(emplManager;(I[Employee] /\ (emplOrgRole /\ -Delta);'Manage

                 (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Ma
                 DELETE FROM emplIssuedEqt[Employee*Equipment]
                  SELECTFROM -(maEmployee~;maEqtKind;eqtKind~) /\ -((emplOrgRole /\ -Delta

                 (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrg
                 DELETE FROM Isn{detyp=Employee}
```

```
        SELECTFROM (-(emplManager;emplManager~) /\ -((emplOrgRole /\ -Delta);'Di

       (TO MAINTAIN  -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Di
       (TO MAINTAIN  -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];
       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplIssuedEqt;eqtKind /\ -((emplOrgRole /\ -Delta);stdIssu

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRol
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;(emplOrgRole /\ -Delta)~ \ (emplIssuedEqt;

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRo
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;(emplOrgRole /\ -Delta)~ \ -(eqtKind~;empl

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRol
       ONE OF DELETE FROM emplManager[Employee*Employee]
               SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager

              (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplO
              DELETE FROM emplManager[Employee*Employee]
               SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager

              (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplO
       (MAINTAINING -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;'
       ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM (-(maEmployee~;maEqtKind) /\ -((emplOrgRole /\ -Delta)

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;(-(maEmployee~;maEqtKind) /\ -((emplOrg

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
(MAINTAINING -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Director'[
(MAINTAINING -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];emplOrgR
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIss
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssu
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (e
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssue
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(e
```

------ Derivation ----->

```
ALL of DELETE FROM emplManager[Employee*Employee]
        SELECTFROM -(emplManager;(I[Employee] /\ (emplOrgRole /\ -Delta);'Manager'[Or

       (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;maEqtKind;eqtKind~) /\ -((emplOrgRole /\ -Delta);std

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-(emplManager;emplManager~) /\ -((emplOrgRole /\ -Delta);'Directo

       (TO MAINTAIN  -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Directo
       (TO MAINTAIN  -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];emplO
       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdI
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIs
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplIssuedEqt;eqtKind /\ -((emplOrgRole /\ -Delta);stdIssueEqtK

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;std
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;(emplOrgRole /\ -Delta)~ \ (emplIssuedEqt;eqtKi

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;(emplOrgRole /\ -Delta)~ \ -(eqtKind~;emplIssue

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       ONE OF DELETE FROM emplManager[Employee*Employee]
               SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/

              (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRol
              DELETE FROM emplManager[Employee*Employee]
               SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/

              (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRol
       (MAINTAINING -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;'Manag
       ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM (-(maEmployee~;maEqtKind) /\ -((emplOrgRole /\ -Delta);stdI

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ empl
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;(-(maEmployee~;maEqtKind) /\ -((emplOrgRole

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ empl
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Director'[Organ
(MAINTAINING -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];emplOrgRole~
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqt
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtK
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (emplIs
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtKi
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(eqtKin
```

<------End Derivation --

```
ON INSERT Delta IN stdIssueEqtKind[OrganizationalRole*EqtKind] EXECUTE    -- (EC
BLOCK
(CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue equip
```

<------End Derivation --

```
ON DELETE Delta FROM stdIssueEqtKind[OrganizationalRole*EqtKind] EXECUTE    -- (
ALL of DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;maEqtKind;eqtKind~) /\ -(emplOrgRole;(stdIssueE

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrg
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;(stdIssueEqtKind /\

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRol
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;(stdIssueEqtK

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((stdIssueEqtKind /\ -Delta)~;emplOrgRole~ \ (emplIssuedEqt;

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRo
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((stdIssueEqtKind /\ -Delta)~;emplOrgRole~ \ -(eqtKind~;empl

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRol
       ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM (-(maEmployee~;maEqtKind) /\ -(emplOrgRole;(stdIssueEq
              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;(-(maEmployee~;maEqtKind) /\ -(emplOrgR

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdI
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdI
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIss
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssue
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (em
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueE
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(e
```

------ Derivation ----->

```
ALL of DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;maEqtKind;eqtKind~) /\ -(emplOrgRole;(stdIssueEqtKin

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;(stdIssueEqtKind /\ -Del

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;std
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;(stdIssueEqtKind /

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdI
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIs
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((stdIssueEqtKind /\ -Delta)~;emplOrgRole~ \ (emplIssuedEqt;eqtKi

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((stdIssueEqtKind /\ -Delta)~;emplOrgRole~ \ -(eqtKind~;emplIssue

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM (-(maEmployee~;maEqtKind) /\ -(emplOrgRole;(stdIssueEqtKind

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ empl
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;(-(maEmployee~;maEqtKind) /\ -(emplOrgRole;(

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ empl
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqt
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtK
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (emplIs
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtKi
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(eqtKin
```

<------End Derivation --

```
ON INSERT Delta IN emplIssuedEqt[Employee*Equipment] EXECUTE    -- (ECA rule 9)
ALL of INSERT INTO Isn{detyp=EqtCompanyID}
        SELECTFROM (eqtID~;(emplIssuedEqt \/ Delta)~;emplIssuedEqt;eqtID /\ eqtI

       (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[Equ
       INSERT INTO eqtApprovedProp[Equipment*Equipment]
        SELECTFROM ((emplIssuedEqt \/ Delta)~;emplIssuedEqt /\ I[Equipment] /\ -

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtAppro
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp;(emplIssuedEqt \/ Delta)~;emplIssuedEqt /\ e

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtAppro
       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipm
       INSERT INTO needsToReturnEqt[Employee*Employee]
        SELECTFROM ((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[

       (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind))
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (emplIssuedEqt;(emplIssuedEqt \/ Delta)~ /\ -I[Employee]) \/

       (TO MAINTAIN  -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ empl
       ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (((emplIssuedEqt \/ Delt
                     THEN INSERT INTO eqtID[Equipment*EqtCompanyID]
                           SELECTFROM 'a'[Equipment]*'b'[EqtCompanyID]

                          (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equi
                     PICK a,b FROM eqtID~;(((emplIssuedEqt \/ Delta)~;emplIssued
                     THEN INSERT INTO eqtID[Equipment*EqtCompanyID]
                           SELECTFROM 'b'[Equipment]*'a'[EqtCompanyID]

                          (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equi
              (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eq
              NEW x:EqtCompanyID;
                INSERT INTO eqtID[Equipment*EqtCompanyID]
                 SELECTFROM (((emplIssuedEqt \/ Delta)~;emplIssuedEqt /\ I[Equip

                (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/
              (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eq
       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqt
       ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt /\ -(maE
                     THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                           SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                          (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;
                     PICK a,b FROM maEmployee;((emplIssuedEqt /\ -(maEmployee~;ma
                     THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[
                                        THEN INSERT INTO maEqtKind[ManagerApprova
                                              SELECTFROM 'a'[ManagerApproval]*'b

                                             (TO MAINTAIN  -emplIssuedEqt \/ maE
                                        PICK a,b FROM maEqtKind~;('a'[ManagerAppr
                                        THEN INSERT INTO eqtKind[Equipment*EqtKi
                                              SELECTFROM 'b'[Equipment]*'a'[EqtK

                                             (TO MAINTAIN  -emplIssuedEqt \/ maE
                                 (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEq
                                 NEW x:EqtKind;
                                   ALL of INSERT INTO maEqtKind[ManagerApproval*
                                           SELECTFROM 'a'[ManagerApproval]*'b'[E

                                          (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                                          INSERT INTO eqtKind[Equipment*EqtKind]
                                           SELECTFROM 'b'[Equipment]*'a'[Manager

                                          (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                                   (MAINTAINING -emplIssuedEqt \/ maEmployee~;ma
                                 (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEq
                          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;e
              (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ e
              NEW x:ManagerApproval;
                ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                        SELECTFROM 'x'[ManagerApproval]*((emplIssuedEqt~ /\ -(eq

                       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqt
                       ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Man
                                     THEN INSERT INTO maEqtKind[ManagerApproval*
                                           SELECTFROM 'a'[ManagerApproval]*'b'[E

                                          (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                                     PICK a,b FROM maEqtKind~;('x'[ManagerApprov
                                     THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                           SELECTFROM 'b'[Equipment]*'a'[EqtKind

                                          (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                              (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKi
                              NEW x:EqtKind;
                                ALL of INSERT INTO maEqtKind[ManagerApproval*Eqt
                                        SELECTFROM 'x'[ManagerApproval]*((emplIs
                                (TO MAINTAIN  -emplIssuedEqt \/ maEmploye
                                INSERT INTO eqtKind[Equipment*EqtKind]
                                 SELECTFROM ((emplIssuedEqt~ /\ -(eqtKind

                                (TO MAINTAIN  -emplIssuedEqt \/ maEmploye
                         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqt
                       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKi
                (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKi
         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/
       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ e
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt /\ -(maEm
              THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                    SELECTFROM ’a’[Employee]*’b’[OrganizationalRole]

                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;
              PICK a,b FROM emplOrgRole~;((emplIssuedEqt /\ -(maEmployee~
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’a’[
                                 THEN INSERT INTO stdIssueEqtKind[Organiza
                                       SELECTFROM ’a’[OrganizationalRole]

                                      (TO MAINTAIN  -emplIssuedEqt \/ maEm
                                 PICK a,b FROM stdIssueEqtKind~;(’a’[Orga
                                 THEN INSERT INTO eqtKind[Equipment*EqtKi
                                       SELECTFROM ’b’[Equipment]*’a’[EqtK

                                      (TO MAINTAIN  -emplIssuedEqt \/ maEm
                          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEq
                          NEW x:EqtKind;
                            ALL of INSERT INTO stdIssueEqtKind[Organizati
                                    SELECTFROM ’a’[OrganizationalRole]*’b

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                                   INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM ’b’[Equipment]*’a’[Organiza

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                            (MAINTAINING -emplIssuedEqt \/ maEmployee~;ma
                          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEq
                   (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;e
       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ e
       NEW x:OrganizationalRole;
         ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                 SELECTFROM ((emplIssuedEqt /\ -(maEmployee~;maEqtKind;eq

                (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqt
                ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’x’[Org
                              THEN INSERT INTO stdIssueEqtKind[Organizati
                                    SELECTFROM ’a’[OrganizationalRole]*’b
```

```
                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                              PICK a,b FROM stdIssueEqtKind~;('x'[Organiz
                              THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM 'b'[Equipment]*'a'[EqtKind

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmpl
                       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKi
                       NEW x:EqtKind;
                         ALL of INSERT INTO stdIssueEqtKind[Organizational
                                 SELECTFROM 'x'[OrganizationalRole]*((empl

                                (TO MAINTAIN  -emplIssuedEqt \/ maEmployee
                                INSERT INTO eqtKind[Equipment*EqtKind]
                                 SELECTFROM ((emplIssuedEqt~ /\ -(eqtKind

                                (TO MAINTAIN  -emplIssuedEqt \/ maEmployee
                         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtK
                       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKi
                (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtK
         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/
       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ em
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgR
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;
              PICK a,b FROM maEmployee;((emplIssuedEqt;eqtKind /\ -(maEmp
              THEN INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                    SELECTFROM 'a'[ManagerApproval]*'b'[EqtKind]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       NEW x:ManagerApproval;
         ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                 SELECTFROM 'x'[ManagerApproval]*((eqtKind~;emplIssuedEqt

                (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maE
                INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                 SELECTFROM 'x'[ManagerApproval]*((emplIssuedEqt;eqtKind

                (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maE
         (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind
              THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                    SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;
              PICK a,b FROM emplOrgRole~;((emplIssuedEqt;eqtKind /\ -(maEm
                     THEN INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                           SELECTFROM 'a'[OrganizationalRole]*'b'[EqtKind]

                          (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
              NEW x:OrganizationalRole;
                ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                        SELECTFROM ((emplIssuedEqt;eqtKind /\ -(maEmployee~;maEq

                       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maE
                       INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                        SELECTFROM 'x'[OrganizationalRole]*((emplIssuedEqt;eqtKi

                       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maE
                (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FRO
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FRO
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
(MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKi
(MAINTAINING -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ emplIssuedE
```

------ Derivation ----->

```
ALL of INSERT INTO Isn{detyp=EqtCompanyID}
        SELECTFROM (eqtID~;(emplIssuedEqt \/ Delta)~;emplIssuedEqt;eqtID /\ eqtID~;eq

       (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[Equipmen
       INSERT INTO eqtApprovedProp[Equipment*Equipment]
        SELECTFROM ((emplIssuedEqt \/ Delta)~;emplIssuedEqt /\ I[Equipment] /\ -eqtAp

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedPr
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp;(emplIssuedEqt \/ Delta)~;emplIssuedEqt /\ eqtApp

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtApprovedPr
       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipment];
       INSERT INTO needsToReturnEqt[Employee*Employee]
        SELECTFROM ((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKi

       (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[Eq
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (emplIssuedEqt;(emplIssuedEqt \/ Delta)~ /\ -I[Employee]) \/ (Delt
(TO MAINTAIN  -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ emplIssu
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (((emplIssuedEqt \/ Delta)~;e
              THEN INSERT INTO eqtID[Equipment*EqtCompanyID]
                    SELECTFROM ’a’[Equipment]*’b’[EqtCompanyID]

                   (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment
              PICK a,b FROM eqtID~;(((emplIssuedEqt \/ Delta)~;emplIssuedEqt /
              THEN INSERT INTO eqtID[Equipment*EqtCompanyID]
                    SELECTFROM ’b’[Equipment]*’a’[EqtCompanyID]

                   (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment
       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;e
       NEW x:EqtCompanyID;
         INSERT INTO eqtID[Equipment*EqtCompanyID]
          SELECTFROM (((emplIssuedEqt \/ Delta)~;emplIssuedEqt /\ I[Equipment]

         (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtI
       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;e
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ F
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt /\ -(maEmploy
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM ’b’[ManagerApproval]*’a’[Employee]

                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKi
              PICK a,b FROM maEmployee;((emplIssuedEqt /\ -(maEmployee~;maEqtK
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’a’[Manag
                                 THEN INSERT INTO maEqtKind[ManagerApproval*Eq
                                       SELECTFROM ’a’[ManagerApproval]*’b’[Eqt

                                      (TO MAINTAIN  -emplIssuedEqt \/ maEmploy
                                 PICK a,b FROM maEqtKind~;(’a’[ManagerApproval
                                 THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                       SELECTFROM ’b’[Equipment]*’a’[EqtKind]

                                      (TO MAINTAIN  -emplIssuedEqt \/ maEmploy
                          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind
                          NEW x:EqtKind;
                            ALL of INSERT INTO maEqtKind[ManagerApproval*EqtKi
                                    SELECTFROM ’a’[ManagerApproval]*’b’[Equipm

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                                   INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM ’b’[Equipment]*’a’[ManagerAppro

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                            (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKi
                          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind
                   (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKin
       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOr
NEW x:ManagerApproval;
  ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
          SELECTFROM 'x'[ManagerApproval]*((emplIssuedEqt~ /\ -(eqtKind

         (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~
         ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[ManagerA
                       THEN INSERT INTO maEqtKind[ManagerApproval*EqtKi
                             SELECTFROM 'a'[ManagerApproval]*'b'[EqtKin

                            (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                       PICK a,b FROM maEqtKind~;('x'[ManagerApproval]*(
                       THEN INSERT INTO eqtKind[Equipment*EqtKind]
                             SELECTFROM 'b'[Equipment]*'a'[EqtKind]

                            (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eq
                NEW x:EqtKind;
                  ALL of INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                          SELECTFROM 'x'[ManagerApproval]*((emplIssuedE

                         (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;ma
                         INSERT INTO eqtKind[Equipment*EqtKind]
                          SELECTFROM ((emplIssuedEqt~ /\ -(eqtKind;maEq

                         (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;ma
                  (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;
                (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eq
         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~
  (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ empl
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOr
ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt /\ -(maEmploy
       THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
             SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

            (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKi
       PICK a,b FROM emplOrgRole~;((emplIssuedEqt /\ -(maEmployee~;maEq
       THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[Organ
                          THEN INSERT INTO stdIssueEqtKind[Organization
                                SELECTFROM 'a'[OrganizationalRole]*'b'[

                               (TO MAINTAIN  -emplIssuedEqt \/ maEmploy
                          PICK a,b FROM stdIssueEqtKind~;('a'[Organizat
                          THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                SELECTFROM 'b'[Equipment]*'a'[EqtKind]

                               (TO MAINTAIN  -emplIssuedEqt \/ maEmploy
                   (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind
                   NEW x:EqtKind;
                     ALL of INSERT INTO stdIssueEqtKind[OrganizationalR
                             SELECTFROM 'a'[OrganizationalRole]*'b'[Equ
```

```
                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                                   INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM 'b'[Equipment]*'a'[Organization

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                            (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKi
                          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind
                   (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKin
       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOr
       NEW x:OrganizationalRole;
         ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                 SELECTFROM ((emplIssuedEqt /\ -(maEmployee~;maEqtKind;eqtKind

                (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~
                ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Organiza
                              THEN INSERT INTO stdIssueEqtKind[OrganizationalR
                                    SELECTFROM 'a'[OrganizationalRole]*'b'[Eqt

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                              PICK a,b FROM stdIssueEqtKind~;('x'[Organization
                              THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM 'b'[Equipment]*'a'[EqtKind]

                                   (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~
                       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eq
                       NEW x:EqtKind;
                         ALL of INSERT INTO stdIssueEqtKind[OrganizationalRole
                                 SELECTFROM 'x'[OrganizationalRole]*((emplIssu

                                (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;ma
                                INSERT INTO eqtKind[Equipment*EqtKind]
                                 SELECTFROM ((emplIssuedEqt~ /\ -(eqtKind;maEq

                                (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;ma
                         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;
                       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eq
                (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~
         (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ empl
       (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOr
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;s
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind /\ -(
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqt
              PICK a,b FROM maEmployee;((emplIssuedEqt;eqtKind /\ -(maEmployee
              THEN INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                    SELECTFROM 'a'[ManagerApproval]*'b'[EqtKind]
```

```
                          (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqt
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplO
              NEW x:ManagerApproval;
                ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                        SELECTFROM 'x'[ManagerApproval]*((eqtKind~;emplIssuedEqt~ /\

                       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKin
                       INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                        SELECTFROM 'x'[ManagerApproval]*((emplIssuedEqt;eqtKind /\ -(

                       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKin
                (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emp
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplO
              ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind /\ -(
                     THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                           SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

                          (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqt
                     PICK a,b FROM emplOrgRole~;((emplIssuedEqt;eqtKind /\ -(maEmploy
                     THEN INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                           SELECTFROM 'a'[OrganizationalRole]*'b'[EqtKind]

                          (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqt
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplO
              NEW x:OrganizationalRole;
                ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                        SELECTFROM ((emplIssuedEqt;eqtKind /\ -(maEmployee~;maEqtKind

                       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKin
                       INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                        SELECTFROM 'x'[OrganizationalRole]*((emplIssuedEqt;eqtKind /\

                       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKin
                (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emp
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplO
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FROM Iss
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FROM Iss
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKind*Em
(MAINTAINING -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ emplIssuedEqt::E
```

<------End Derivation --

```
ON DELETE Delta FROM emplIssuedEqt[Employee*Equipment] EXECUTE    -- (ECA rule 1
ALL of DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -(((emplIssuedEqt /\ -Delta);eqtKind /\ -(emplOrgRole;stdIssu

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRol
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKi

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ ((emplIssuedEqt /\ -Delta);

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRo
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ -(eqtKind~;(emplIssuedEqt /

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRol
       ONE OF DELETE FROM emplOrgRole[Employee*OrganizationalRole]
               SELECTFROM (-(emplOwnsEqt;eqtKind) /\ -((emplIssuedEqt /\ -Delta)

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKin
              DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM emplOrgRole~;(-(emplOwnsEqt;eqtKind) /\ -((emplIssuedE

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKin
       (MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emp
       ONE OF DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -((emplIs

              (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/
              DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -((emplIs

              (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/
              DELETE FROM Isn{detyp=EqtKind}
               SELECTFROM -(eqtKind~;(I[Equipment] /\ -((emplIssuedEqt /\ -Delta

              (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/
       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind
(MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssued
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Eq
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIss
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssu
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (e
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssue
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(e
```

------ Derivation ----->

```
ALL of DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -(((emplIssuedEqt /\ -Delta);eqtKind /\ -(emplOrgRole;stdIssueEqtK

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;std
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKind;-(

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdI
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIs
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ ((emplIssuedEqt /\ -Delta);eqtKi

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ -(eqtKind~;(emplIssuedEqt /\ -De

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       ONE OF DELETE FROM emplOrgRole[Employee*OrganizationalRole]
               SELECTFROM (-(emplOwnsEqt;eqtKind) /\ -((emplIssuedEqt /\ -Delta);eqtK

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/
              DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM emplOrgRole~;(-(emplOwnsEqt;eqtKind) /\ -((emplIssuedEqt /\

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/
       (MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssu
       ONE OF DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -((emplIssuedE

              (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtK
              DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -((emplIssuedE

              (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtK
              DELETE FROM Isn{detyp=EqtKind}
               SELECTFROM -(eqtKind~;(I[Equipment] /\ -((emplIssuedEqt /\ -Delta)~;(e

              (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtK
       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[
(MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssuedEqt;e
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Equipme
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqt
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtK
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (emplIs
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtKi
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(eqtKin
```

<------End Derivation --

```
ON INSERT Delta IN emplOwnsEqt[Employee*Equipment] EXECUTE    -- (ECA rule 11)
BLOCK
(CANNOT CHANGE V[Equipment*Equipment] FROM Coherence of registered equipment)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Equipment*Equipment] FROM Coherence of registered equipment)
```

<------End Derivation --

```
ON DELETE Delta FROM emplOwnsEqt[Employee*Equipment] EXECUTE    -- (ECA rule 12)
ALL of DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -((emplOwnsEqt /\ -Delta)~;(emplOwnsEqt /\ -Delta)) /\ -(eqtI

       (TO MAINTAIN  -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ F
       ONE OF DELETE FROM emplOrgRole[Employee*OrganizationalRole]
               SELECTFROM (-((emplOwnsEqt /\ -Delta);eqtKind) /\ -(emplIssuedEqt

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKin
              DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM emplOrgRole~;(-((emplOwnsEqt /\ -Delta);eqtKind) /\ -(

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKin
        (MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ empl
(MAINTAINING -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ FROM Cohe
(MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssued
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -((emplOwnsEqt /\ -Delta)~;(emplOwnsEqt /\ -Delta)) /\ -(eqtID;eqt

       (TO MAINTAIN  -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ FROM C
       ONE OF DELETE FROM emplOrgRole[Employee*OrganizationalRole]
               SELECTFROM (-((emplOwnsEqt /\ -Delta);eqtKind) /\ -(emplIssuedEqt;eqtK

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/
              DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
               SELECTFROM emplOrgRole~;(-((emplOwnsEqt /\ -Delta);eqtKind) /\ -(emplI

              (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/
        (MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssu
(MAINTAINING -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ FROM Coherence
(MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssuedEqt;e
```

<------End Derivation --

```
ON INSERT Delta IN eqtMake[Equipment*EqtMake] EXECUTE    -- (ECA rule 13)
ONE OF INSERT INTO Isn{detyp=EqtMake}
        SELECTFROM ((eqtMake \/ Delta)~;eqtMake /\ -I[EqtMake]) \/ ((eqtMake \/ I

       (TO MAINTAIN  -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipme
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=EqtMake}
        SELECTFROM (Delta~;Delta /\ I[EqtMake]) - I[EqtMake]

(MAINTAINING -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*EqtMa
(MAINTAINING -I[Equipment] \/ eqtMake;eqtMake~ FROM TOT eqtMake::Equipment*EqtMa
```

------ Derivation ----->

```
ONE OF INSERT INTO Isn{detyp=EqtMake}
        SELECTFROM ((eqtMake \/ Delta)~;eqtMake /\ -I[EqtMake]) \/ ((eqtMake \/ Delta

       (TO MAINTAIN  -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*Eq
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=EqtMake}
        SELECTFROM (Delta~;Delta /\ I[EqtMake]) - I[EqtMake]

(MAINTAINING -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*EqtMake)
(MAINTAINING -I[Equipment] \/ eqtMake;eqtMake~ FROM TOT eqtMake::Equipment*EqtMake)
```

<------End Derivation --

```
ON DELETE Delta FROM eqtMake[Equipment*EqtMake] EXECUTE    -- (ECA rule 14)
DELETE FROM Isn{detyp=Equipment}
 SELECTFROM -((eqtMake /\ -Delta);(eqtMake /\ -Delta)~) /\ I[Equipment]

(TO MAINTAIN  -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*EqtMa
(TO MAINTAIN  -I[Equipment] \/ eqtMake;eqtMake~ FROM TOT eqtMake::Equipment*EqtMa
```

------ Derivation ----->

```
DELETE FROM Isn{detyp=Equipment}
 SELECTFROM -((eqtMake /\ -Delta);(eqtMake /\ -Delta)~) /\ I[Equipment]
```

```
(TO MAINTAIN  -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*EqtMake)
(TO MAINTAIN  -I[Equipment] \/ eqtMake;eqtMake~ FROM TOT eqtMake::Equipment*EqtMake)
```

<------End Derivation --

```
ON INSERT Delta IN eqtType[Equipment*EqtType] EXECUTE    -- (ECA rule 15)
ONE OF INSERT INTO Isn{detyp=EqtType}
        SELECTFROM ((eqtType \/ Delta)~;eqtType /\ -I[EqtType]) \/ ((eqtType \/ I

       (TO MAINTAIN  -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipmer
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta~;Delta /\ I[EqtType]) - I[EqtType]

(MAINTAINING -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*EqtTy
(MAINTAINING -I[Equipment] \/ eqtType;eqtType~ FROM TOT eqtType::Equipment*EqtTy
```

------ Derivation ----->

```
ONE OF INSERT INTO Isn{detyp=EqtType}
        SELECTFROM ((eqtType \/ Delta)~;eqtType /\ -I[EqtType]) \/ ((eqtType \/ Delta

       (TO MAINTAIN  -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*Eq
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta~;Delta /\ I[EqtType]) - I[EqtType]

(MAINTAINING -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*EqtType)
(MAINTAINING -I[Equipment] \/ eqtType;eqtType~ FROM TOT eqtType::Equipment*EqtType)
```

<------End Derivation --

```
ON DELETE Delta FROM eqtType[Equipment*EqtType] EXECUTE    -- (ECA rule 16)
DELETE FROM Isn{detyp=Equipment}
 SELECTFROM -((eqtType /\ -Delta);(eqtType /\ -Delta)~) /\ I[Equipment]

(TO MAINTAIN  -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*EqtTy
(TO MAINTAIN  -I[Equipment] \/ eqtType;eqtType~ FROM TOT eqtType::Equipment*EqtTy
```

------ Derivation ----->

```
DELETE FROM Isn{detyp=Equipment}
 SELECTFROM -((eqtType /\ -Delta);(eqtType /\ -Delta)~) /\ I[Equipment]

(TO MAINTAIN  -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*EqtType)
(TO MAINTAIN  -I[Equipment] \/ eqtType;eqtType~ FROM TOT eqtType::Equipment*EqtType)
```

<------End Derivation --

```
     ON INSERT Delta IN eqtSerial[Equipment*EqtSerial] EXECUTE    -- (ECA rule 17)
     ONE OF INSERT INTO Isn{detyp=EqtSerial}
             SELECTFROM ((eqtSerial \/ Delta)~;eqtSerial /\ -I[EqtSerial]) \/ ((eqtSe

            (TO MAINTAIN  -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial:
            INSERT INTO Isn{detyp=Equipment}
             SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

            INSERT INTO Isn{detyp=EqtSerial}
             SELECTFROM (Delta~;Delta /\ I[EqtSerial]) - I[EqtSerial]

     (MAINTAINING -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equipme
     (MAINTAINING -I[Equipment] \/ eqtSerial;eqtSerial~ FROM TOT eqtSerial::Equipment
```

------ Derivation ----->

```
ONE OF INSERT INTO Isn{detyp=EqtSerial}
        SELECTFROM ((eqtSerial \/ Delta)~;eqtSerial /\ -I[EqtSerial]) \/ ((eqtSerial

       (TO MAINTAIN  -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equi
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=EqtSerial}
        SELECTFROM (Delta~;Delta /\ I[EqtSerial]) - I[EqtSerial]

(MAINTAINING -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equipment*Eq
(MAINTAINING -I[Equipment] \/ eqtSerial;eqtSerial~ FROM TOT eqtSerial::Equipment*EqtS
```

<------End Derivation --

```
     ON DELETE Delta FROM eqtSerial[Equipment*EqtSerial] EXECUTE    -- (ECA rule 18)
     DELETE FROM Isn{detyp=Equipment}
      SELECTFROM -((eqtSerial /\ -Delta);(eqtSerial /\ -Delta)~) /\ I[Equipment]

     (TO MAINTAIN  -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equipm
     (TO MAINTAIN  -I[Equipment] \/ eqtSerial;eqtSerial~ FROM TOT eqtSerial::Equipmen
```

------ Derivation ----->

```
DELETE FROM Isn{detyp=Equipment}
 SELECTFROM -((eqtSerial /\ -Delta);(eqtSerial /\ -Delta)~) /\ I[Equipment]

(TO MAINTAIN  -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equipment*E
(TO MAINTAIN  -I[Equipment] \/ eqtSerial;eqtSerial~ FROM TOT eqtSerial::Equipment*Eqt
```

<------End Derivation --

```
     ON INSERT Delta IN eqtKind[Equipment*EqtKind] EXECUTE    -- (ECA rule 19)
     ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind /\ -(ma
                   THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                         SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                        (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKi
                   PICK a,b FROM maEmployee;((emplIssuedEqt;eqtKind /\ -(maEmployee~;
                   THEN INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                         SELECTFROM 'a'[ManagerApproval]*'b'[EqtKind]

                        (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKi
            (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
            NEW x:ManagerApproval;
              ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                      SELECTFROM 'x'[ManagerApproval]*((eqtKind~;emplIssuedEqt~ /\ -(

                     (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind
                     INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                      SELECTFROM 'x'[ManagerApproval]*((emplIssuedEqt;eqtKind /\ -(ma

                     (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplO
            (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
            ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind /\ -(ma
                   THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                         SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

                        (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKi
                   PICK a,b FROM emplOrgRole~;((emplIssuedEqt;eqtKind /\ -(maEmployee
                   THEN INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                         SELECTFROM 'a'[OrganizationalRole]*'b'[EqtKind]

                        (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKi
            (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
            NEW x:OrganizationalRole;
              ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                      SELECTFROM ((emplIssuedEqt;eqtKind /\ -(maEmployee~;maEqtKind)
                     (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \
                     INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                      SELECTFROM 'x'[OrganizationalRole]*((emplIssuedEqt;eqtKind /\ -

                     (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \
              (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplO
           (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
            INSERT INTO needsToReturnEqt[Employee*Employee]
             SELECTFROM ((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[

            (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind))
            INSERT INTO Isn{detyp=EqtKind}
             SELECTFROM ((eqtKind \/ Delta)~;eqtKind /\ -I[EqtKind]) \/ ((eqtKind \/ 

            (TO MAINTAIN  -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipme
            INSERT INTO Isn{detyp=Equipment}
             SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

            INSERT INTO Isn{detyp=EqtKind}
             SELECTFROM (Delta~;Delta /\ I[EqtKind]) - I[EqtKind]

     (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
     (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKi
     (MAINTAINING -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*EqtKi
     (MAINTAINING -I[Equipment] \/ eqtKind;eqtKind~ FROM TOT eqtKind::Equipment*EqtKi
```

------ Derivation ----->

```
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind /\ -(maEmplo
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
              PICK a,b FROM maEmployee;((emplIssuedEqt;eqtKind /\ -(maEmployee~;maEqt
              THEN INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                    SELECTFROM 'a'[ManagerApproval]*'b'[EqtKind]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
       NEW x:ManagerApproval;
         ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                 SELECTFROM 'x'[ManagerApproval]*((eqtKind~;emplIssuedEqt~ /\ -(maEqt

                (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ em
                INSERT INTO maEqtKind[ManagerApproval*EqtKind]
                 SELECTFROM 'x'[ManagerApproval]*((emplIssuedEqt;eqtKind /\ -(maEmplo
                (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ em
         (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRol
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((emplIssuedEqt;eqtKind /\ -(maEmplo
              THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                    SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
              PICK a,b FROM emplOrgRole~;((emplIssuedEqt;eqtKind /\ -(maEmployee~;maE
              THEN INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                    SELECTFROM 'a'[OrganizationalRole]*'b'[EqtKind]

                   (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
       NEW x:OrganizationalRole;
         ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                 SELECTFROM ((emplIssuedEqt;eqtKind /\ -(maEmployee~;maEqtKind) /\ -(

                (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ em
                INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                 SELECTFROM 'x'[OrganizationalRole]*((emplIssuedEqt;eqtKind /\ -(maEm

                (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ em
         (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRol
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
       INSERT INTO needsToReturnEqt[Employee*Employee]
        SELECTFROM ((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKi

       (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[Eq
       INSERT INTO Isn{detyp=EqtKind}
        SELECTFROM ((eqtKind \/ Delta)~;eqtKind /\ -I[EqtKind]) \/ ((eqtKind \/ Delta

       (TO MAINTAIN  -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*Eq
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=EqtKind}
        SELECTFROM (Delta~;Delta /\ I[EqtKind]) - I[EqtKind]

(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKind*Em
(MAINTAINING -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*EqtKind)
(MAINTAINING -I[Equipment] \/ eqtKind;eqtKind~ FROM TOT eqtKind::Equipment*EqtKind)
```

<------End Derivation --

```
ON DELETE Delta FROM eqtKind[Equipment*EqtKind] EXECUTE    -- (ECA rule 20)
ONE OF DELETE FROM emplOrgRole[Employee*OrganizationalRole]
```

```
        SELECTFROM (-(emplOwnsEqt;(eqtKind /\ -Delta)) /\ -(emplIssuedEqt;(eqtKi

       (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emp
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM emplOrgRole~;(-(emplOwnsEqt;(eqtKind /\ -Delta)) /\ -(emplIss

       (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emp
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;maEqtKind;(eqtKind /\ -Delta)~) /\ -(emplOrgRol

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrg
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM stdIssueEqtKind;(-((eqtKind~ /\ -Delta~);(I[Equipment] /\ -(em

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKin
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM stdIssueEqtKind;(-((eqtKind /\ -Delta)~;(I[Equipment] /\ -(emp

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKin
       DELETE FROM Isn{detyp=EqtKind}
        SELECTFROM -((eqtKind /\ -Delta)~;(I[Equipment] /\ -(emplIssuedEqt~;empl

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKin
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplIssuedEqt;(eqtKind /\ -Delta) /\ -(emplOrgRole;stdIssu

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRol
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKin

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ (emplIssuedEqt;(eqtKind /\

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRol
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -noNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKind

       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ -((eqtKind /\ -Delta)~;empl

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRol
       DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -((eqtKind /\ -Delta);(eqtKind /\ -Delta)~) /\ I[Equipment]

       (TO MAINTAIN  -I[Equipment] \/ eqtKind;I[EqtKind];eqtKind~ FROM UNI eqtKi
(MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssued
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Eq
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIss
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssu
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (e
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssue
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(e
(MAINTAINING -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*EqtKi
(MAINTAINING -I[Equipment] \/ eqtKind;eqtKind~ FROM TOT eqtKind::Equipment*EqtKi
```

------ Derivation ----->

```
ONE OF DELETE FROM emplOrgRole[Employee*OrganizationalRole]
        SELECTFROM (-(emplOwnsEqt;(eqtKind /\ -Delta)) /\ -(emplIssuedEqt;(eqtKind /\

       (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIss
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM emplOrgRole~;(-(emplOwnsEqt;(eqtKind /\ -Delta)) /\ -(emplIssuedEq

       (TO MAINTAIN  -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIss
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;maEqtKind;(eqtKind /\ -Delta)~) /\ -(emplOrgRole;std

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM stdIssueEqtKind;(-((eqtKind~ /\ -Delta~);(I[Equipment] /\ -(emplIs

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM stdIssueEqtKind;(-((eqtKind /\ -Delta)~;(I[Equipment] /\ -(emplIss

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I
       DELETE FROM Isn{detyp=EqtKind}
        SELECTFROM -((eqtKind /\ -Delta)~;(I[Equipment] /\ -(emplIssuedEqt~;emplIssue

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplIssuedEqt;(eqtKind /\ -Delta) /\ -(emplOrgRole;stdIssueEqtK

       (TO MAINTAIN  -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;std
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKind;-(e

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdI
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ (emplIssuedEqt;(eqtKind /\ -Delt

       (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -noNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKind;(eqt
       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIs
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(stdIssueEqtKind~;emplOrgRole~ \ -((eqtKind /\ -Delta)~;emplIssue

       (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \
       DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -((eqtKind /\ -Delta);(eqtKind /\ -Delta)~) /\ I[Equipment]

       (TO MAINTAIN  -I[Equipment] \/ eqtKind;I[EqtKind];eqtKind~ FROM UNI eqtKind::E
(MAINTAINING -(emplOrgRole;stdIssueEqtKind) \/ emplOwnsEqt;eqtKind \/ emplIssuedEqt;e
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Equipme
(MAINTAINING -needsToReturnEqt \/ (emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqt
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtK
(MAINTAINING -allNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ (emplIs
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtKi
(MAINTAINING -noNecessaryEqtHasBeenIssued \/ stdIssueEqtKind~;emplOrgRole~ \ -(eqtKin
(MAINTAINING -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*EqtKind)
(MAINTAINING -I[Equipment] \/ eqtKind;eqtKind~ FROM TOT eqtKind::Equipment*EqtKind)
```

<------End Derivation --

```
          ON INSERT Delta IN eqtStatus[Equipment*EqtStatus] EXECUTE    -- (ECA rule 21)
          ONE OF INSERT INTO Isn{detyp=EqtStatus}
                  SELECTFROM ((eqtStatus \/ Delta)~;eqtStatus /\ -I[EqtStatus]) \/ ((eqtSt

                 (TO MAINTAIN  -(eqtStatus~;eqtStatus) \/ I[EqtStatus] FROM UNI eqtStatus:
                 INSERT INTO Isn{detyp=Equipment}
                  SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

                 INSERT INTO Isn{detyp=EqtStatus}
                  SELECTFROM (Delta~;Delta /\ I[EqtStatus]) - I[EqtStatus]

          (MAINTAINING -(eqtStatus~;eqtStatus) \/ I[EqtStatus] FROM UNI eqtStatus::Equipme
          (MAINTAINING -I[Equipment] \/ eqtStatus;eqtStatus~ FROM TOT eqtStatus::Equipment
```

------ Derivation ----->

```
     ONE OF INSERT INTO Isn{detyp=EqtStatus}
             SELECTFROM ((eqtStatus \/ Delta)~;eqtStatus /\ -I[EqtStatus]) \/ ((eqtStatus

            (TO MAINTAIN  -(eqtStatus~;eqtStatus) \/ I[EqtStatus] FROM UNI eqtStatus::Equi
            INSERT INTO Isn{detyp=Equipment}
             SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

            INSERT INTO Isn{detyp=EqtStatus}
```

```
                SELECTFROM (Delta~;Delta /\ I[EqtStatus]) - I[EqtStatus]

        (MAINTAINING -(eqtStatus~;eqtStatus) \/ I[EqtStatus] FROM UNI eqtStatus::Equipment*Eq
        (MAINTAINING -I[Equipment] \/ eqtStatus;eqtStatus~ FROM TOT eqtStatus::Equipment*EqtS
```

<------End Derivation --

```
             ON DELETE Delta FROM eqtStatus[Equipment*EqtStatus] EXECUTE    -- (ECA rule 22)
             DELETE FROM Isn{detyp=Equipment}
              SELECTFROM -((eqtStatus /\ -Delta);(eqtStatus /\ -Delta)~) /\ I[Equipment]

             (TO MAINTAIN  -(eqtStatus~;eqtStatus) \/ I[EqtStatus] FROM UNI eqtStatus::Equipme
             (TO MAINTAIN  -I[Equipment] \/ eqtStatus;eqtStatus~ FROM TOT eqtStatus::Equipmen
```

------ Derivation ----->

```
        DELETE FROM Isn{detyp=Equipment}
         SELECTFROM -((eqtStatus /\ -Delta);(eqtStatus /\ -Delta)~) /\ I[Equipment]

        (TO MAINTAIN  -(eqtStatus~;eqtStatus) \/ I[EqtStatus] FROM UNI eqtStatus::Equipment*E
        (TO MAINTAIN  -I[Equipment] \/ eqtStatus;eqtStatus~ FROM TOT eqtStatus::Equipment*Eqt
```

<------End Derivation --

```
             ON INSERT Delta IN eqtID[Equipment*EqtCompanyID] EXECUTE    -- (ECA rule 23)
             BLOCK
             (CANNOT CHANGE V[Equipment*Equipment] FROM Coherence of registered equipment)
```

------ Derivation ----->

```
        BLOCK
        (CANNOT CHANGE V[Equipment*Equipment] FROM Coherence of registered equipment)
```

<------End Derivation --

```
             ON DELETE Delta FROM eqtID[Equipment*EqtCompanyID] EXECUTE    -- (ECA rule 24)
             ALL of ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
                            SELECTFROM emplIssuedEqt;(-((eqtID /\ -Delta);(eqtID~ /\ -Delta~)

                           (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ e
                           DELETE FROM emplIssuedEqt[Employee*Equipment]
                            SELECTFROM emplIssuedEqt;(-((eqtID /\ -Delta);(eqtID /\ -Delta)~)
                           (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ e
                           DELETE FROM Isn{detyp=Equipment}
                            SELECTFROM -((eqtID /\ -Delta);(eqtID /\ -Delta)~) /\ emplIssuedE

                           (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ e
                    (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqt
                    DELETE FROM Isn{detyp=Equipment}
                     SELECTFROM -(emplOwnsEqt~;emplOwnsEqt) /\ -((eqtID /\ -Delta);(eqtID /\

                    (TO MAINTAIN  -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ F
             (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FRO
             (MAINTAINING -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ FROM Cohe
```

------ Derivation ----->

```
ALL of ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;(-((eqtID /\ -Delta);(eqtID~ /\ -Delta~)) /\

              (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;
              DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;(-((eqtID /\ -Delta);(eqtID /\ -Delta)~) /\ e

              (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;
              DELETE FROM Isn{detyp=Equipment}
               SELECTFROM -((eqtID /\ -Delta);(eqtID /\ -Delta)~) /\ emplIssuedEqt~;e

              (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;
       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ F
       DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -(emplOwnsEqt~;emplOwnsEqt) /\ -((eqtID /\ -Delta);(eqtID /\ -Delt

       (TO MAINTAIN  -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ FROM C
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FROM Iss
(MAINTAINING -I[Equipment] \/ emplOwnsEqt~;emplOwnsEqt \/ eqtID;eqtID~ FROM Coherence
```

<------End Derivation --

```
     ON INSERT Delta IN maEmployee[ManagerApproval*Employee] EXECUTE    -- (ECA rule
     BLOCK
     (CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue equip
```

<------End Derivation --

```
          ON DELETE Delta FROM maEmployee[ManagerApproval*Employee] EXECUTE    -- (ECA rul
          ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
                  SELECTFROM -((maEmployee /\ -Delta)~;maEqtKind;eqtKind~) /\ -(emplOrgRol

                 (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrg
                 DELETE FROM emplIssuedEqt[Employee*Equipment]
                  SELECTFROM (-((maEmployee /\ -Delta)~;maEqtKind) /\ -(emplOrgRole;stdIss

                 (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOr
                 DELETE FROM eqtKind[Equipment*EqtKind]
                  SELECTFROM emplIssuedEqt~;(-((maEmployee /\ -Delta)~;maEqtKind) /\ -(emp

                 (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOr
                 DELETE FROM maManager[ManagerApproval*Employee]
                  SELECTFROM -((maEmployee /\ -Delta);emplManager) /\ maManager

                 (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager approval 
                 DELETE FROM Isn{detyp=ManagerApproval}
                  SELECTFROM -((maEmployee /\ -Delta);emplManager;maManager~) /\ I[Manager

                 (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FR
                 DELETE FROM Isn{detyp=ManagerApproval}
                  SELECTFROM -((maEmployee /\ -Delta);(maEmployee /\ -Delta)~) /\ I[Manage

                 (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;I[Employee];maEmployee~ F
          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
          (MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
          (MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
          (MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
          (MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
          (MAINTAINING -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::Manag
          (MAINTAINING -I[ManagerApproval] \/ maEmployee;maEmployee~ FROM TOT maEmployee::
```

------ Derivation ----->

```
     ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
             SELECTFROM -((maEmployee /\ -Delta)~;maEqtKind;eqtKind~) /\ -(emplOrgRole;std

            (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;
            DELETE FROM emplIssuedEqt[Employee*Equipment]
             SELECTFROM (-((maEmployee /\ -Delta)~;maEqtKind) /\ -(emplOrgRole;stdIssueEqt

            (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole
            DELETE FROM eqtKind[Equipment*EqtKind]
             SELECTFROM emplIssuedEqt~;(-((maEmployee /\ -Delta)~;maEqtKind) /\ -(emplOrgR
       (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole
       DELETE FROM maManager[ManagerApproval*Employee]
        SELECTFROM -((maEmployee /\ -Delta);emplManager) /\ maManager

       (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager approval integ
       DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -((maEmployee /\ -Delta);emplManager;maManager~) /\ I[ManagerAppro

       (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM Ma
       DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -((maEmployee /\ -Delta);(maEmployee /\ -Delta)~) /\ I[ManagerAppr

       (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;I[Employee];maEmployee~ FROM U
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::ManagerApp
(MAINTAINING -I[ManagerApproval] \/ maEmployee;maEmployee~ FROM TOT maEmployee::Manag
```

<------End Derivation --

```
ON INSERT Delta IN maManager[ManagerApproval*Employee] EXECUTE    -- (ECA rule 2
ONE OF INSERT INTO emplManager[Employee*Employee]
        SELECTFROM (maEmployee~;maManager /\ -emplManager) \/ (maEmployee~;Delta

       (TO MAINTAIN  -(maEmployee~;maManager) \/ emplManager FROM Manager approva
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (emplManager~;maEmployee~;maManager /\ -I[Employee]) \/ (empl

       (TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM M
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((maManager /\ -(maEmployee;empl
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manage
              PICK a,b FROM maEmployee~;((maManager /\ -(maEmployee;emplManager)
              THEN INSERT INTO emplManager[Employee*Employee]
                    SELECTFROM 'a'[Employee]*'b'[Employee]

                   (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manage
       (MAINTAINING -maManager \/ maEmployee;emplManager FROM Manager approval i
       NEW x:Employee;
         ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                 SELECTFROM ((maManager /\ -(maEmployee;emplManager)) \/ (Delta
                (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager a
                INSERT INTO emplManager[Employee*Employee]
                 SELECTFROM 'x'[Employee]*((maManager /\ -(maEmployee;emplManage

                (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager a
         (MAINTAINING -maManager \/ maEmployee;emplManager FROM Manager approval
       (MAINTAINING -maManager \/ maEmployee;emplManager FROM Manager approval i
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM ((maManager \/ Delta)~;maManager /\ -I[Employee]) \/ ((maMana

       (TO MAINTAIN  -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::
       INSERT INTO Isn{detyp=ManagerApproval}
        SELECTFROM (Delta;Delta~ /\ I[ManagerApproval]) - I[ManagerApproval]

       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (Delta~;Delta /\ I[Employee]) - I[Employee]

(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerA
(MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::Mana
```

------ Derivation ----->

```
ONE OF INSERT INTO emplManager[Employee*Employee]
        SELECTFROM (maEmployee~;maManager /\ -emplManager) \/ (maEmployee~;Delta /\ -

       (TO MAINTAIN  -(maEmployee~;maManager) \/ emplManager FROM Manager approval in
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (emplManager~;maEmployee~;maManager /\ -I[Employee]) \/ (emplManag

       (TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM Manage
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((maManager /\ -(maEmployee;emplMana
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager ap
              PICK a,b FROM maEmployee~;((maManager /\ -(maEmployee;emplManager)) \/
              THEN INSERT INTO emplManager[Employee*Employee]
                    SELECTFROM 'a'[Employee]*'b'[Employee]

                   (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager ap
       (MAINTAINING -maManager \/ maEmployee;emplManager FROM Manager approval integr
       NEW x:Employee;
         ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                 SELECTFROM ((maManager /\ -(maEmployee;emplManager)) \/ (Delta /\ -(
                (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager appro
                INSERT INTO emplManager[Employee*Employee]
                 SELECTFROM 'x'[Employee]*((maManager /\ -(maEmployee;emplManager)) \

                (TO MAINTAIN  -maManager \/ maEmployee;emplManager FROM Manager appro
         (MAINTAINING -maManager \/ maEmployee;emplManager FROM Manager approval inte
       (MAINTAINING -maManager \/ maEmployee;emplManager FROM Manager approval integr
       INSERT INTO Isn{detyp=Employee}
        SELECTFROM ((maManager \/ Delta)~;maManager /\ -I[Employee]) \/ ((maManager \

       (TO MAINTAIN  -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::Manag
       INSERT INTO Isn{detyp=ManagerApproval}
        SELECTFROM (Delta;Delta~ /\ I[ManagerApproval]) - I[ManagerApproval]

       INSERT INTO Isn{detyp=Employee}
        SELECTFROM (Delta~;Delta /\ I[Employee]) - I[Employee]

(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerApprov
(MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::ManagerA
```

<------End Derivation --

```
ON DELETE Delta FROM maManager[ManagerApproval*Employee] EXECUTE    -- (ECA rule
ONE OF DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -(maEmployee;emplManager;(maManager /\ -Delta)~) /\ I[Manager

       (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FR
       DELETE FROM maEmployee[ManagerApproval*Employee]
        SELECTFROM -((maManager /\ -Delta);emplManager~) /\ maEmployee

       (TO MAINTAIN  -maEmployee~ \/ emplManager;maManager~ FROM Manager approva
       DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -((maManager /\ -Delta);(maManager /\ -Delta)~) /\ I[ManagerA

       (TO MAINTAIN  -I[ManagerApproval] \/ maManager;I[Employee];maManager~ FRO
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerA
(MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::Mana
```

------ Derivation ----->

```
ONE OF DELETE FROM Isn{detyp=ManagerApproval}
```

```
        SELECTFROM -(maEmployee;emplManager;(maManager /\ -Delta)~) /\ I[ManagerAppro

       (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM Ma
       DELETE FROM maEmployee[ManagerApproval*Employee]
        SELECTFROM -((maManager /\ -Delta);emplManager~) /\ maEmployee

       (TO MAINTAIN  -maEmployee~ \/ emplManager;maManager~ FROM Manager approval int
       DELETE FROM Isn{detyp=ManagerApproval}
        SELECTFROM -((maManager /\ -Delta);(maManager /\ -Delta)~) /\ I[ManagerApprov

       (TO MAINTAIN  -I[ManagerApproval] \/ maManager;I[Employee];maManager~ FROM UNI
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerApprov
(MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::ManagerA
```

<------End Derivation --

```
ON INSERT Delta IN maEqtKind[ManagerApproval*EqtKind] EXECUTE    -- (ECA rule 29
BLOCK
(CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Employee*EqtKind] FROM No manager approvals for standard issue equip
```

<------End Derivation --

```
ON DELETE Delta FROM maEqtKind[ManagerApproval*EqtKind] EXECUTE    -- (ECA rule 
ALL of DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;(maEqtKind /\ -Delta);eqtKind~) /\ -(emplOrgRol

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrg
       ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM (-(maEmployee~;(maEqtKind /\ -Delta)) /\ -(emplOrgRole

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;(-(maEmployee~;(maEqtKind /\ -Delta)) /

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrg
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;std
```

------ Derivation ----->

```
ALL of DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM -(maEmployee~;(maEqtKind /\ -Delta);eqtKind~) /\ -(emplOrgRole;std

       (TO MAINTAIN  -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;
       ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM (-(maEmployee~;(maEqtKind /\ -Delta)) /\ -(emplOrgRole;stdI

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ empl
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;(-(maEmployee~;(maEqtKind /\ -Delta)) /\ -(e

              (TO MAINTAIN  -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ empl
       (MAINTAINING -(emplIssuedEqt;eqtKind) \/ maEmployee~;maEqtKind \/ emplOrgRole;
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
(MAINTAINING -emplIssuedEqt \/ maEmployee~;maEqtKind;eqtKind~ \/ emplOrgRole;stdIssue
```

<------End Derivation --

```
ON INSERT Delta IN eqtApprovedProp[Equipment*Equipment] EXECUTE    -- (ECA rule
ONE OF INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp \/ Delta)~;emplIssuedEqt~;emplIssuedEqt /\ (

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtAppro
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM emplIssuedEqt~;emplIssuedEqt;(eqtApprovedProp \/ Delta)~ /\ (

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipm
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp \/ Delta)~;emplOwnsEqt~;emplOwnsEqt /\ (eqtA

       (TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedP
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM emplOwnsEqt~;emplOwnsEqt;(eqtApprovedProp \/ Delta)~ /\ (eqtA

       (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment]
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp \/ Delta)~;eqtApprovedBySecOff;'Yes'[Yes/No

       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer]
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp /\ -I[Equipment]) \/ (Delta /\ -I[Equipment]
       (TO MAINTAIN  -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
       DELETE FROM eqtSecReqt[Equipment*SecRequirement]
        SELECTFROM ((eqtApprovedProp /\ -(eqtSecReqt~ \ eqtSatReqt~) /\ -(eqtAppr

       (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprove
       INSERT INTO eqtSatReqt[Equipment*SecRequirement]
        SELECTFROM ((eqtApprovedProp~ /\ -(eqtSatReqt / eqtSecReqt) /\ -(eqtAppro

       (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprove
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((eqtApprovedProp /\ -(eqtSecRe
              THEN INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No answer]
                    SELECTFROM ’a’[Equipment]*’b’[Yes/No answer]

                   (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \
              PICK a,b FROM eqtApprovedBySecOff~;((eqtApprovedProp /\ -(eqtSecRe
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’a’[Yes/No a
                                 THEN BLOCK
                                      (CANNOT CHANGE ’Yes’[Yes/No answer] FROM de
                                 PICK a,b FROM ’Yes’[Yes/No answer];(’a’[Yes/No a
                                 THEN INSERT INTO eqtApprovedBySecOff[Equipment*
                                       SELECTFROM ’b’[Equipment]*’a’[Yes/No answ

                                      (TO MAINTAIN  -eqtApprovedProp \/ eqtSecRe
                          (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatRe
                          NEW x:Yes/No answer;
                            ALL of BLOCK
                                   (CANNOT CHANGE ’Yes’[Yes/No answer] FROM dele
                                   INSERT INTO eqtApprovedBySecOff[Equipment*Yes/
                                    SELECTFROM ’b’[Equipment]*’a’[Yes/No answer]*

                                   (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~
                            (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSat
                          (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatRe
                   (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/
       (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApproved
       NEW x:Yes/No answer;
         ALL of INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No answer]
                 SELECTFROM ((eqtApprovedProp /\ -(eqtSecReqt~ \ eqtSatReqt~) /\

                (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ e
                ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’x’[Yes/No answer]*((
                       THEN BLOCK
                            (CANNOT CHANGE ’Yes’[Yes/No answer] FROM deleqtAppro
                       PICK a,b FROM ’Yes’[Yes/No answer];(’x’[Yes/No answer]*((
                       THEN INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No ans
                             SELECTFROM ’b’[Equipment]*’a’[Yes/No answer]

                            (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtS
                (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eq
                (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprov
              (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedI
              INSERT INTO Isn{detyp=Equipment}
               SELECTFROM (eqtApprovedProp;eqtApprovedProp /\ -I[Equipment]) \/ (eqtAppr

              (TO MAINTAIN  -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI
              INSERT INTO Isn{detyp=Equipment}
               SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

              INSERT INTO Isn{detyp=Equipment}
               SELECTFROM (Delta~;Delta /\ I[Equipment]) - I[Equipment]

       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp I
       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp I
       (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
       (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
       (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I
       (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I
       (MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
       (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOff
       (MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM ASY eqtApprovedProp::Equipment
       (MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtAppro
       (MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM INJ eqtAppro
```

------ Derivation ----->

```
ONE OF INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp \/ Delta)~;emplIssuedEqt~;emplIssuedEqt /\ (eqtAp

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtApprovedPr
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM emplIssuedEqt~;emplIssuedEqt;(eqtApprovedProp \/ Delta)~ /\ (eqtAp

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipment];
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp \/ Delta)~;emplOwnsEqt~;emplOwnsEqt /\ (eqtApprov

       (TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedProp~;
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM emplOwnsEqt~;emplOwnsEqt;(eqtApprovedProp \/ Delta)~ /\ (eqtApprov

       (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment];eqtA
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp \/ Delta)~;eqtApprovedBySecOff;'Yes'[Yes/No answe

       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtA
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;(eqt
(TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;
INSERT INTO Isn{detyp=Equipment}
 SELECTFROM (eqtApprovedProp /\ -I[Equipment]) \/ (Delta /\ -I[Equipment])

(TO MAINTAIN  -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
DELETE FROM eqtSecReqt[Equipment*SecRequirement]
 SELECTFROM ((eqtApprovedProp /\ -(eqtSecReqt~ \ eqtSatReqt~) /\ -(eqtApproved

(TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySe
INSERT INTO eqtSatReqt[Equipment*SecRequirement]
 SELECTFROM ((eqtApprovedProp~ /\ -(eqtSatReqt / eqtSecReqt) /\ -(eqtApprovedB

(TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySe
ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((eqtApprovedProp /\ -(eqtSecReqt~ \
       THEN INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No answer]
             SELECTFROM 'a'[Equipment]*'b'[Yes/No answer]

            (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqt
       PICK a,b FROM eqtApprovedBySecOff~;((eqtApprovedProp /\ -(eqtSecReqt~ \
       THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[Yes/No answe
                          THEN BLOCK
                               (CANNOT CHANGE 'Yes'[Yes/No answer] FROM deleqt
                          PICK a,b FROM 'Yes'[Yes/No answer];('a'[Yes/No answe
                          THEN INSERT INTO eqtApprovedBySecOff[Equipment*Yes/N
                                SELECTFROM 'b'[Equipment]*'a'[Yes/No answer]

                               (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \
                   (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~
                   NEW x:Yes/No answer;
                     ALL of BLOCK
                            (CANNOT CHANGE 'Yes'[Yes/No answer] FROM deleqtApp
                            INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No a
                             SELECTFROM 'b'[Equipment]*'a'[Yes/No answer]*'x'[

                            (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eq
                     (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt
                   (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~
            (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtA
(MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySec
NEW x:Yes/No answer;
  ALL of INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No answer]
          SELECTFROM ((eqtApprovedProp /\ -(eqtSecReqt~ \ eqtSatReqt~) /\ -(eq

         (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApp
         ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Yes/No answer]*((eqtAp
                THEN BLOCK
                     (CANNOT CHANGE 'Yes'[Yes/No answer] FROM deleqtApprovedPr
                PICK a,b FROM 'Yes'[Yes/No answer];('x'[Yes/No answer]*((eqtAp
                THEN INSERT INTO eqtApprovedBySecOff[Equipment*Yes/No answer]
```

```
                             SELECTFROM 'b'[Equipment]*'a'[Yes/No answer]

                            (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReq
                (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtAppr
         (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedByS
       (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySec
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp;eqtApprovedProp /\ -I[Equipment]) \/ (eqtApproved

       (TO MAINTAIN  -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtA
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta~;Delta /\ I[Equipment]) - I[Equipment]

(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
(MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM Equi
(MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM Equi
(MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
(MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
(MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
(MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOff;'Ye
(MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM ASY eqtApprovedProp::Equipment*Equ
(MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtApprovedP
(MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM INJ eqtApprovedP
```

<------End Derivation --

```
    ON DELETE Delta FROM eqtApprovedProp[Equipment*Equipment] EXECUTE    -- (ECA rul
    ALL of ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
                   SELECTFROM emplIssuedEqt;((-eqtApprovedProp~ /\ emplIssuedEqt~;em

                  (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ e
                  DELETE FROM emplIssuedEqt[Employee*Equipment]
                   SELECTFROM emplIssuedEqt;((-eqtApprovedProp /\ emplIssuedEqt~;empl

                  (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ e
                  DELETE FROM Isn{detyp=Equipment}
                   SELECTFROM (-eqtApprovedProp /\ emplIssuedEqt~;emplIssuedEqt /\ I

                  (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ e
           (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprov
           ONE OF DELETE FROM emplOwnsEqt[Employee*Equipment]
                   SELECTFROM emplOwnsEqt;((-eqtApprovedProp~ /\ emplOwnsEqt~;emplOw

                  (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtAp
                  DELETE FROM emplOwnsEqt[Employee*Equipment]
                   SELECTFROM emplOwnsEqt;((-eqtApprovedProp /\ emplOwnsEqt~;emplOwns

                  (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtAp
                  DELETE FROM Isn{detyp=Equipment}
                   SELECTFROM (-eqtApprovedProp /\ emplOwnsEqt~;emplOwnsEqt /\ I[Equ

                  (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtAp
           (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedPr
           ONE OF DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
                   SELECTFROM ((-eqtApprovedProp /\ eqtApprovedBySecOff;'Yes'[Yes/No

                  (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprov
                  DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
                   SELECTFROM ((-eqtApprovedProp~ /\ eqtApprovedBySecOff;'Yes'[Yes/N

                  (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprov
                  DELETE FROM Isn{detyp=Equipment}
                   SELECTFROM (-eqtApprovedProp /\ eqtApprovedBySecOff;'Yes'[Yes/No 

                  (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprov
           (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecO
           DELETE FROM Isn{detyp=Equipment}
            SELECTFROM (-eqtApprovedProp /\ -(eqtSecReqt;-eqtSatReqt~) /\ I[Equipmen

           (TO MAINTAIN  -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~
    (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp 
    (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
    (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ 
    (MAINTAINING -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM in
```

------ Derivation ----->

```
ALL of ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;((-eqtApprovedProp~ /\ emplIssuedEqt~;emplIss

              (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApp
              DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;((-eqtApprovedProp /\ emplIssuedEqt~;emplIssu

              (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApp
              DELETE FROM Isn{detyp=Equipment}
               SELECTFROM (-eqtApprovedProp /\ emplIssuedEqt~;emplIssuedEqt /\ I[Equi

              (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApp
       (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedPro
       ONE OF DELETE FROM emplOwnsEqt[Employee*Equipment]
               SELECTFROM emplOwnsEqt;((-eqtApprovedProp~ /\ emplOwnsEqt~;emplOwnsEqt
              (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprove
              DELETE FROM emplOwnsEqt[Employee*Equipment]
               SELECTFROM emplOwnsEqt;((-eqtApprovedProp /\ emplOwnsEqt~;emplOwnsEqt

              (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprove
              DELETE FROM Isn{detyp=Equipment}
               SELECTFROM (-eqtApprovedProp /\ emplOwnsEqt~;emplOwnsEqt /\ I[Equipmen

              (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprove
       (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FR
       ONE OF DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
               SELECTFROM ((-eqtApprovedProp /\ eqtApprovedBySecOff;'Yes'[Yes/No answ

              (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedByS
              DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
               SELECTFROM ((-eqtApprovedProp~ /\ eqtApprovedBySecOff;'Yes'[Yes/No ans

              (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedByS
              DELETE FROM Isn{detyp=Equipment}
               SELECTFROM (-eqtApprovedProp /\ eqtApprovedBySecOff;'Yes'[Yes/No answe

              (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedByS
       (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /
       DELETE FROM Isn{detyp=Equipment}
        SELECTFROM (-eqtApprovedProp /\ -(eqtSecReqt;-eqtSatReqt~) /\ I[Equipment]) \

       (TO MAINTAIN  -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
(MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM Equi
(MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
(MAINTAINING -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM inseqtA
```

<------End Derivation --

```
     ON INSERT Delta IN eqtSecReqt[Equipment*SecRequirement] EXECUTE    -- (ECA rule 
     ALL of INSERT INTO Isn{detyp=Equipment}
             SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

            INSERT INTO Isn{detyp=SecRequirement}
             SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

------ Derivation ----->

```
ALL of INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]
```

```
            INSERT INTO Isn{detyp=SecRequirement}
             SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

<------End Derivation --

```
          ON DELETE Delta FROM eqtSecReqt[Equipment*SecRequirement] EXECUTE    -- (ECA rul
          ALL of DELETE FROM Isn{detyp=Equipment}
                  SELECTFROM -eqtApprovedProp /\ -((eqtSecReqt /\ -Delta);-eqtSatReqt~) /\

                 (TO MAINTAIN  -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~
                 DELETE FROM eqtApprovedProp[Equipment*Equipment]
                  SELECTFROM -((eqtSecReqt /\ -Delta)~ \ eqtSatReqt~) /\ -(eqtApprovedBySe

                 (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprove
          (MAINTAINING -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM in
          (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOf
```

------ Derivation ----->

```
     ALL of DELETE FROM Isn{detyp=Equipment}
             SELECTFROM -eqtApprovedProp /\ -((eqtSecReqt /\ -Delta);-eqtSatReqt~) /\ I[Eq

            (TO MAINTAIN  -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM
            DELETE FROM eqtApprovedProp[Equipment*Equipment]
             SELECTFROM -((eqtSecReqt /\ -Delta)~ \ eqtSatReqt~) /\ -(eqtApprovedBySecOff;

            (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySe
     (MAINTAINING -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM inseqtA
     (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOff;'Ye
```

<------End Derivation --

```
          ON INSERT Delta IN eqtSatReqt[Equipment*SecRequirement] EXECUTE    -- (ECA rule
          ALL of INSERT INTO Isn{detyp=Equipment}
                  SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

                 INSERT INTO Isn{detyp=SecRequirement}
                  SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

------ Derivation ----->

```
ALL of INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (Delta;Delta~ /\ I[Equipment]) - I[Equipment]

       INSERT INTO Isn{detyp=SecRequirement}
        SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

<------End Derivation --

```
ON DELETE Delta FROM eqtSatReqt[Equipment*SecRequirement] EXECUTE    -- (ECA rul
ALL of DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -eqtApprovedProp /\ -(eqtSecReqt;-(eqtSatReqt /\ -Delta)~) /\

       (TO MAINTAIN  -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -(eqtSecReqt~ \ (eqtSatReqt /\ -Delta)~) /\ -(eqtApprovedBySe

       (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprove
(MAINTAINING -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM in
(MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOf
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=Equipment}
        SELECTFROM -eqtApprovedProp /\ -(eqtSecReqt;-(eqtSatReqt /\ -Delta)~) /\ I[Eq

       (TO MAINTAIN  -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -(eqtSecReqt~ \ (eqtSatReqt /\ -Delta)~) /\ -(eqtApprovedBySecOff;

       (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySe
(MAINTAINING -I[Equipment] \/ eqtApprovedProp \/ eqtSecReqt;-eqtSatReqt~ FROM inseqtA
(MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOff;'Ye
```

<------End Derivation --

```
ON INSERT Delta IN eqtApprovedBySecOff[Equipment*Yes/No answer] EXECUTE    -- (E
ALL of INSERT INTO eqtApprovedProp[Equipment*Equipment]
        SELECTFROM (eqtApprovedBySecOff;'Yes'[Yes/No answer];(eqtApprovedBySecOf

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Yes/No answer];(eq
       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;’Yes’[Yes/No answer]
       (TO MAINTAIN  -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySec
       INSERT INTO Isn{detyp=Yes/No answer}
        SELECTFROM (Delta~;Delta /\ I[Yes/No answer]) - I[Yes/No answer]

(MAINTAINING -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~ /\
(MAINTAINING -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~ /\
(MAINTAINING -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~ /\
```

------ Derivation ----->

```
  ALL of INSERT INTO eqtApprovedProp[Equipment*Equipment]
          SELECTFROM (eqtApprovedBySecOff;’Yes’[Yes/No answer];(eqtApprovedBySecOff \/

         (TO MAINTAIN  -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~
         INSERT INTO Isn{detyp=Equipment}
          SELECTFROM (eqtApprovedProp;eqtApprovedBySecOff;’Yes’[Yes/No answer];(eqtAppr

         (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtA
         (TO MAINTAIN  -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~;
         INSERT INTO Isn{detyp=Yes/No answer}
          SELECTFROM (Delta~;Delta /\ I[Yes/No answer]) - I[Yes/No answer]

  (MAINTAINING -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
  (MAINTAINING -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
  (MAINTAINING -(eqtApprovedBySecOff;’Yes’[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
```

<------End Derivation --

```
       ON DELETE Delta FROM eqtApprovedBySecOff[Equipment*Yes/No answer] EXECUTE    --
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -(eqtSecReqt~ \ eqtSatReqt~) /\ -((eqtApprovedBySecOff /\ -Delta);’Y

       (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecO
```

------ Derivation ----->

```
  DELETE FROM eqtApprovedProp[Equipment*Equipment]
   SELECTFROM -(eqtSecReqt~ \ eqtSatReqt~) /\ -((eqtApprovedBySecOff /\ -Delta);’Yes’[Y

  (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOff;’Y
```

<------End Derivation --

```
ON INSERT Delta IN typeApprovedProp[EqtType*EqtType] EXECUTE    -- (ECA rule 39)
ONE OF INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp \/ Delta)~;typeApprovedBySecOff;'Yes'[Yes/No

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp /\ -I[EqtType]) \/ (Delta /\ -I[EqtType])

       (TO MAINTAIN  -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
       DELETE FROM typeSecReqt[EqtType*SecRequirement]
        SELECTFROM ((typeApprovedProp /\ -(typeSecReqt~ \ typeSatReqt~) /\ -(type

       (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppr
       INSERT INTO typeSatReqt[EqtType*SecRequirement]
        SELECTFROM ((typeApprovedProp~ /\ -(typeSatReqt / typeSecReqt) /\ -(type

       (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppr
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((typeApprovedProp /\ -(typeSecR
              THEN INSERT INTO typeApprovedBySecOff[EqtType*Yes/No answer]
                    SELECTFROM 'a'[EqtType]*'b'[Yes/No answer]

                   (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt
              PICK a,b FROM typeApprovedBySecOff~;((typeApprovedProp /\ -(typeSec
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[Yes/No a
                                 THEN BLOCK
                                      (CANNOT CHANGE 'Yes'[Yes/No answer] FROM de
                                 PICK a,b FROM 'Yes'[Yes/No answer];('a'[Yes/No a
                                 THEN INSERT INTO typeApprovedBySecOff[EqtType*Ye
                                       SELECTFROM 'b'[EqtType]*'a'[Yes/No answer]

                                      (TO MAINTAIN  -typeApprovedProp \/ typeSecR
                          (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSa
                          NEW x:Yes/No answer;
                            ALL of BLOCK
                                   (CANNOT CHANGE 'Yes'[Yes/No answer] FROM delt
                                   INSERT INTO typeApprovedBySecOff[EqtType*Yes/N
                                    SELECTFROM 'b'[EqtType]*'a'[Yes/No answer]*'

                                   (TO MAINTAIN  -typeApprovedProp \/ typeSecReq
                            (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ type
                          (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSa
                   (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~
       (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppro
       NEW x:Yes/No answer;
         ALL of INSERT INTO typeApprovedBySecOff[EqtType*Yes/No answer]
```

```
                 SELECTFROM ((typeApprovedProp /\ -(typeSecReqt~ \ typeSatReqt~)

                (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \
                ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Yes/No answer]*((
                       THEN BLOCK
                            (CANNOT CHANGE 'Yes'[Yes/No answer] FROM deltypeAppro
                       PICK a,b FROM 'Yes'[Yes/No answer];('x'[Yes/No answer]*((
                       THEN INSERT INTO typeApprovedBySecOff[EqtType*Yes/No answ
                             SELECTFROM 'b'[EqtType]*'a'[Yes/No answer]

                            (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ ty
                (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/
         (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAp
      (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppr
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp;typeApprovedProp /\ -I[EqtType]) \/ (typeAp

       (TO MAINTAIN  -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta;Delta~ /\ I[EqtType]) - I[EqtType]

       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta~;Delta /\ I[EqtType]) - I[EqtType]

(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBySe
(MAINTAINING -typeApprovedProp \/ I[EqtType] FROM ASY typeApprovedProp::EqtType*
(MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI typeAppr
(MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM INJ typeAppr
```

------ Derivation ----->

```
ONE OF INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp \/ Delta)~;typeApprovedBySecOff;'Yes'[Yes/No ans

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~;(t

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp /\ -I[EqtType]) \/ (Delta /\ -I[EqtType])

       (TO MAINTAIN  -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
       DELETE FROM typeSecReqt[EqtType*SecRequirement]
        SELECTFROM ((typeApprovedProp /\ -(typeSecReqt~ \ typeSatReqt~) /\ -(typeAppr
(TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApproved
INSERT INTO typeSatReqt[EqtType*SecRequirement]
 SELECTFROM ((typeApprovedProp~ /\ -(typeSatReqt / typeSecReqt) /\ -(typeAppro

(TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApproved
ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((typeApprovedProp /\ -(typeSecReqt~
       THEN INSERT INTO typeApprovedBySecOff[EqtType*Yes/No answer]
             SELECTFROM ’a’[EqtType]*’b’[Yes/No answer]

            (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/
       PICK a,b FROM typeApprovedBySecOff~;((typeApprovedProp /\ -(typeSecReqt
       THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’a’[Yes/No answe
                          THEN BLOCK
                               (CANNOT CHANGE ’Yes’[Yes/No answer] FROM deltyp
                          PICK a,b FROM ’Yes’[Yes/No answer];(’a’[Yes/No answe
                          THEN INSERT INTO typeApprovedBySecOff[EqtType*Yes/No
                                SELECTFROM ’b’[EqtType]*’a’[Yes/No answer]

                               (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~
                   (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReq
                   NEW x:Yes/No answer;
                     ALL of BLOCK
                            (CANNOT CHANGE ’Yes’[Yes/No answer] FROM deltypeAp
                            INSERT INTO typeApprovedBySecOff[EqtType*Yes/No an
                             SELECTFROM ’b’[EqtType]*’a’[Yes/No answer]*’x’[Ye

                            (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \
                     (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatR
                   (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReq
            (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ t
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedB
NEW x:Yes/No answer;
  ALL of INSERT INTO typeApprovedBySecOff[EqtType*Yes/No answer]
          SELECTFROM ((typeApprovedProp /\ -(typeSecReqt~ \ typeSatReqt~) /\ -

         (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typ
         ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’x’[Yes/No answer]*((typeA
                THEN BLOCK
                     (CANNOT CHANGE ’Yes’[Yes/No answer] FROM deltypeApprovedP
                PICK a,b FROM ’Yes’[Yes/No answer];(’x’[Yes/No answer]*((typeA
                THEN INSERT INTO typeApprovedBySecOff[EqtType*Yes/No answer]
                      SELECTFROM ’b’[EqtType]*’a’[Yes/No answer]

                     (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSat
         (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ type
  (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprove
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedB
INSERT INTO Isn{detyp=EqtType}
 SELECTFROM (typeApprovedProp;typeApprovedProp /\ -I[EqtType]) \/ (typeApprove
       (TO MAINTAIN  -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI type
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta;Delta~ /\ I[EqtType]) - I[EqtType]

       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta~;Delta /\ I[EqtType]) - I[EqtType]

(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
(MAINTAINING -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBySecOff
(MAINTAINING -typeApprovedProp \/ I[EqtType] FROM ASY typeApprovedProp::EqtType*EqtTy
(MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI typeApproved
(MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM INJ typeApproved
```

<------End Derivation --

```
     ON DELETE Delta FROM typeApprovedProp[EqtType*EqtType] EXECUTE    -- (ECA rule 4
     ALL of ONE OF DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
                    SELECTFROM ((-typeApprovedProp /\ typeApprovedBySecOff;'Yes'[Yes/

                   (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeAppr
                   DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
                    SELECTFROM ((-typeApprovedProp~ /\ typeApprovedBySecOff;'Yes'[Yes

                   (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeAppr
                   DELETE FROM Isn{detyp=EqtType}
                    SELECTFROM (-typeApprovedProp /\ typeApprovedBySecOff;'Yes'[Yes/N

                   (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeAppr
            (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
            DELETE FROM Isn{detyp=EqtType}
             SELECTFROM (-typeApprovedProp /\ -(typeSecReqt;-typeSatReqt~) /\ I[EqtTy

            (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt
     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
     (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM i
```

------ Derivation ----->

```
ALL of ONE OF DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
               SELECTFROM ((-typeApprovedProp /\ typeApprovedBySecOff;'Yes'[Yes/No an

              (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedB
              DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
               SELECTFROM ((-typeApprovedProp~ /\ typeApprovedBySecOff;'Yes'[Yes/No a
                   (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedB
                   DELETE FROM Isn{detyp=EqtType}
                    SELECTFROM (-typeApprovedProp /\ typeApprovedBySecOff;'Yes'[Yes/No ans

                   (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedB
            (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~
            DELETE FROM Isn{detyp=EqtType}
             SELECTFROM (-typeApprovedProp /\ -(typeSecReqt;-typeSatReqt~) /\ I[EqtType])

            (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FRO
     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
     (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM instyp
```

<------End Derivation --

```
          ON INSERT Delta IN typeSecReqt[EqtType*SecRequirement] EXECUTE    -- (ECA rule 4
          ALL of INSERT INTO Isn{detyp=EqtType}
                  SELECTFROM (Delta;Delta~ /\ I[EqtType]) - I[EqtType]

                 INSERT INTO Isn{detyp=SecRequirement}
                  SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

------ Derivation ----->

```
     ALL of INSERT INTO Isn{detyp=EqtType}
             SELECTFROM (Delta;Delta~ /\ I[EqtType]) - I[EqtType]

            INSERT INTO Isn{detyp=SecRequirement}
             SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

<------End Derivation --

```
          ON DELETE Delta FROM typeSecReqt[EqtType*SecRequirement] EXECUTE    -- (ECA rule
          ALL of DELETE FROM Isn{detyp=EqtType}
                  SELECTFROM -typeApprovedProp /\ -((typeSecReqt /\ -Delta);-typeSatReqt~)

                 (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt
                 DELETE FROM typeApprovedProp[EqtType*EqtType]
                  SELECTFROM -((typeSecReqt /\ -Delta)~ \ typeSatReqt~) /\ -(typeApprovedB

                 (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppr
          (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM i
          (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedByS
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=EqtType}
        SELECTFROM -typeApprovedProp /\ -((typeSecReqt /\ -Delta);-typeSatReqt~) /\ I

       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FRO
       DELETE FROM typeApprovedProp[EqtType*EqtType]
        SELECTFROM -((typeSecReqt /\ -Delta)~ \ typeSatReqt~) /\ -(typeApprovedBySecO

       (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApproved
(MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM instyp
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBySecOff
```

<------End Derivation --

```
ON INSERT Delta IN typeSatReqt[EqtType*SecRequirement] EXECUTE    -- (ECA rule 4
ALL of INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta;Delta~ /\ I[EqtType]) - I[EqtType]

       INSERT INTO Isn{detyp=SecRequirement}
        SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

------ Derivation ----->

```
ALL of INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (Delta;Delta~ /\ I[EqtType]) - I[EqtType]

       INSERT INTO Isn{detyp=SecRequirement}
        SELECTFROM (Delta~;Delta /\ I[SecRequirement]) - I[SecRequirement]
```

<------End Derivation --

```
ON DELETE Delta FROM typeSatReqt[EqtType*SecRequirement] EXECUTE    -- (ECA rule
ALL of DELETE FROM Isn{detyp=EqtType}
        SELECTFROM -typeApprovedProp /\ -(typeSecReqt;-(typeSatReqt /\ -Delta)~)

       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt
       DELETE FROM typeApprovedProp[EqtType*EqtType]
        SELECTFROM -(typeSecReqt~ \ (typeSatReqt /\ -Delta)~) /\ -(typeApprovedB

       (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppr
(MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM i
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedByS
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=EqtType}
        SELECTFROM -typeApprovedProp /\ -(typeSecReqt;-(typeSatReqt /\ -Delta)~) /\ I

       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FRO
       DELETE FROM typeApprovedProp[EqtType*EqtType]
        SELECTFROM -(typeSecReqt~ \ (typeSatReqt /\ -Delta)~) /\ -(typeApprovedBySecO

       (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApproved
(MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM instyp
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBySecOff
```

<------End Derivation --

```
ON INSERT Delta IN typeApprovedBySecOff[EqtType*Yes/No answer] EXECUTE    -- (EC
ALL of INSERT INTO typeApprovedProp[EqtType*EqtType]
        SELECTFROM (typeApprovedBySecOff;'Yes'[Yes/No answer];(typeApprovedBySec

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp;typeApprovedBySecOff;'Yes'[Yes/No answer];(

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answe
       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
       INSERT INTO Isn{detyp=Yes/No answer}
        SELECTFROM (Delta~;Delta /\ I[Yes/No answer]) - I[Yes/No answer]

(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
```

------ Derivation ----->

```
ALL of INSERT INTO typeApprovedProp[EqtType*EqtType]
        SELECTFROM (typeApprovedBySecOff;'Yes'[Yes/No answer];(typeApprovedBySecOff \

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp;typeApprovedBySecOff;'Yes'[Yes/No answer];(typeA

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
       INSERT INTO Isn{detyp=Yes/No answer}
        SELECTFROM (Delta~;Delta /\ I[Yes/No answer]) - I[Yes/No answer]
```

```
    (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
    (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
    (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
```

<------End Derivation --

```
        ON DELETE Delta FROM typeApprovedBySecOff[EqtType*Yes/No answer] EXECUTE    -- (
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM -(typeSecReqt~ \ typeSatReqt~) /\ -((typeApprovedBySecOff /\ -Delta)

        (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBy
```

------ Derivation ----->

```
    DELETE FROM typeApprovedProp[EqtType*EqtType]
     SELECTFROM -(typeSecReqt~ \ typeSatReqt~) /\ -((typeApprovedBySecOff /\ -Delta);'Yes

    (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBySecOf
```

<------End Derivation --

```
        ON INSERT Delta IN needsToReturnEqt[Employee*Employee] EXECUTE    -- (ECA rule 4
        BLOCK
        (CANNOT CHANGE 'Grey'[Status] FROM setemplStatusGrey)
```

------ Derivation ----->

```
    BLOCK
    (CANNOT CHANGE 'Grey'[Status] FROM setemplStatusGrey)
```

<------End Derivation --

```
        ON DELETE Delta FROM needsToReturnEqt[Employee*Employee] EXECUTE    -- (ECA rule
        ALL of ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
                       SELECTFROM ((-needsToReturnEqt /\ (emplIssuedEqt;eqtKind /\ -(empl

                      (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEq
                      DELETE FROM eqtKind[Equipment*EqtKind]
                       SELECTFROM emplIssuedEqt~;((-needsToReturnEqt /\ (emplIssuedEqt;e

                      (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEq
                      ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((-needsToReturnEqt /\ (
                             THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                                   SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

                                  (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRol
                             PICK a,b FROM emplOrgRole~;((-needsToReturnEqt /\ (emplIssu
                             THEN INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                                   SELECTFROM 'a'[OrganizationalRole]*'b'[EqtKind]

                                  (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRol
                      (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtK
                      NEW x:OrganizationalRole;
                        ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                                SELECTFROM (((-needsToReturnEqt /\ (emplIssuedEqt;eqtKin

                               (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;s
                               INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                                SELECTFROM 'x'[OrganizationalRole]*(((-needsToReturnEqt /

                               (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;s
                        (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueE
                      (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtK
                      DELETE FROM Isn{detyp=Employee}
                       SELECTFROM (-needsToReturnEqt /\ (emplIssuedEqt;eqtKind /\ -(empl

                      (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEq
               (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));
               ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Black'[Status];emplStatus~) /\ -needsTo

                      (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
                      DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Black'[Status];emplStatus~) /\ -needsTo

                      (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
                      DELETE FROM Isn{detyp=Employee}
                       SELECTFROM (-(emplStatus;'Black'[Status];emplStatus~) /\ -needsTo

                      (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
               (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssue
               ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNeces

                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
                      DELETE FROM Isn{detyp=Employee}
                       SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNeces

                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
               (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus
               DELETE FROM Isn{detyp=Employee}
```

```
        SELECTFROM (-(emplStatus;'Yellow'[Status];emplStatus~) /\ -noNecessaryEq

       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/
(MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKi
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ I[
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green
(MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecess
```

------ Derivation ----->

```
ALL of ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM ((-needsToReturnEqt /\ (emplIssuedEqt;eqtKind /\ -(emplOrgR

              (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind
              DELETE FROM eqtKind[Equipment*EqtKind]
               SELECTFROM emplIssuedEqt~;((-needsToReturnEqt /\ (emplIssuedEqt;eqtKin

              (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind
              ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ((-needsToReturnEqt /\ (emplI
                     THEN INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                           SELECTFROM 'a'[Employee]*'b'[OrganizationalRole]

                          (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;std
                     PICK a,b FROM emplOrgRole~;((-needsToReturnEqt /\ (emplIssuedEqt
                     THEN INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                           SELECTFROM 'a'[OrganizationalRole]*'b'[EqtKind]

                          (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;std
              (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind)
              NEW x:OrganizationalRole;
                ALL of INSERT INTO emplOrgRole[Employee*OrganizationalRole]
                        SELECTFROM (((-needsToReturnEqt /\ (emplIssuedEqt;eqtKind /\

                       (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIss
                       INSERT INTO stdIssueEqtKind[OrganizationalRole*EqtKind]
                        SELECTFROM 'x'[OrganizationalRole]*(((-needsToReturnEqt /\ (e

                       (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIss
                (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKin
              (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind)
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-needsToReturnEqt /\ (emplIssuedEqt;eqtKind /\ -(emplOrgRo

              (TO MAINTAIN  -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind
       (MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[Eqt
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-(emplStatus;'Black'[Status];emplStatus~) /\ -needsToRetur
              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-(emplStatus;'Black'[Status];emplStatus~) /\ -needsToRetur

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;'Black'[Status];emplStatus~) /\ -needsToRetur

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\
       ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNecessaryE

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNecessaryE

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
       (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Gre
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-(emplStatus;'Yellow'[Status];emplStatus~) /\ -noNecessaryEqtHasB

       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNec
(MAINTAINING -((emplIssuedEqt;eqtKind /\ -(emplOrgRole;stdIssueEqtKind));V[EqtKind*Em
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ I[Emplo
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green'[Sta
(MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecessaryEq
```

<------End Derivation --

```
     ON INSERT Delta IN allNecessaryEqtHasBeenIssued[Employee*Employee] EXECUTE    --
     BLOCK
     (CANNOT CHANGE 'Grey'[Status] FROM setemplStatusGrey)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE 'Grey'[Status] FROM setemplStatusGrey)
```

<------End Derivation --

```
     ON DELETE Delta FROM allNecessaryEqtHasBeenIssued[Employee*Employee] EXECUTE    -
     ALL of DELETE FROM Isn{detyp=Employee}
             SELECTFROM (-allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKi
            (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole
            (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/
            ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
                    SELECTFROM (-(emplStatus;'Red'[Status];emplStatus~) /\ -allNecess

                   (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emp
                   DELETE FROM Isn{detyp=Employee}
                    SELECTFROM (-(emplStatus;'Red'[Status];emplStatus~) /\ -allNecess

                   (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emp
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;
            ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
                    SELECTFROM (-(emplStatus;'Orange'[Status];emplStatus~) /\ -noNece

                   (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Or
                   DELETE FROM Isn{detyp=Employee}
                    SELECTFROM (-(emplStatus;'Orange'[Status];emplStatus~) /\ -noNece

                   (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Or
           (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[St
     (MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssu
     (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'[S
     (MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecess
     (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];e
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-allNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKind;-(

       (TO MAINTAIN  -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdI
       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNec
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-(emplStatus;'Red'[Status];emplStatus~) /\ -allNecessaryEq

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStat
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;'Red'[Status];emplStatus~) /\ -allNecessaryEq

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStat
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'
       ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM (-(emplStatus;'Orange'[Status];emplStatus~) /\ -noNecessary

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;'Orange'[Status];emplStatus~) /\ -noNecessary
              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'
       (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status]
(MAINTAINING -I[Employee] \/ allNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtK
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'[Status
(MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecessaryEq
(MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];emplSt
```

<------End Derivation --

```
          ON INSERT Delta IN noNecessaryEqtHasBeenIssued[Employee*Employee] EXECUTE    --
          BLOCK
          (CANNOT CHANGE 'Grey'[Status] FROM setemplStatusGrey)
```

------ Derivation ----->

```
        BLOCK
        (CANNOT CHANGE 'Grey'[Status] FROM setemplStatusGrey)
```

<------End Derivation --

```
          ON DELETE Delta FROM noNecessaryEqtHasBeenIssued[Employee*Employee] EXECUTE    -
          ALL of DELETE FROM Isn{detyp=Employee}
                 SELECTFROM (-noNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKin

                (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;
                (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ 
                ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNeces

                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
                      DELETE FROM Isn{detyp=Employee}
                       SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNeces

                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
                (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus
                ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecess

                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
                      DELETE FROM needsToReturnEqt[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecess

                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
                      DELETE FROM Isn{detyp=Employee}
                       SELECTFROM (-(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecess
                      (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
               (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Empl
               ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
                       SELECTFROM (-(emplStatus;'Orange'[Status];emplStatus~) /\ -noNece

                      (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Ora
                      DELETE FROM Isn{detyp=Employee}
                       SELECTFROM (-(emplStatus;'Orange'[Status];emplStatus~) /\ -noNece

                      (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Ora
               (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Sta
        (MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssue
        (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green
        (MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecess
        (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee])
        (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];em
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-noNecessaryEqtHasBeenIssued /\ -(emplOrgRole;stdIssueEqtKind;eqt

       (TO MAINTAIN  -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIs
       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNec
       ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNecessaryE

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;'Green'[Status];emplStatus~) /\ -noNecessaryE

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
       (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Gre
       ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessaryEq

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM (-(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessaryEq

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessaryEq

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
       (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee]
       ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
```

```
               SELECTFROM (-(emplStatus;’Orange’[Status];emplStatus~) /\ -noNecessary

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;’Orange’
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM (-(emplStatus;’Orange’[Status];emplStatus~) /\ -noNecessary

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;’Orange’
       (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;’Orange’[Status]
(MAINTAINING -I[Employee] \/ noNecessaryEqtHasBeenIssued \/ emplOrgRole;stdIssueEqtKi
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;’Green’[Sta
(MAINTAINING -I[Employee] \/ emplStatus;’Yellow’[Status];emplStatus~ \/ noNecessaryEq
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee]) \/ em
(MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;’Orange’[Status];emplSt
```

<------End Derivation --

```
ON INSERT Delta IN emplStatus[Employee*Status] EXECUTE    -- (ECA rule 53)
BLOCK
(CANNOT CHANGE ’Grey’[Status] FROM setemplStatusGrey)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE ’Grey’[Status] FROM setemplStatusGrey)
```

<------End Derivation --

```
ON DELETE Delta FROM emplStatus[Employee*Status] EXECUTE    -- (ECA rule 54)
ALL of ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);’Black’[Status];(emplStatus /

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);’Black’[Status];(emplStatus /

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -((emplStatus /\ -Delta);’Black’[Status];(emplStatus /

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssue
       ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);’Green’[Status];(emplStatus /

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -((emplStatus /\ -Delta);'Green'[Status];(emplStatus /

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
       (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);'Red'[Status];(emplStatus /\

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emp
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -((emplStatus /\ -Delta);'Red'[Status];(emplStatus /\

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emp
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -((emplStatus /\ -Delta);'Yellow'[Status];(emplStatus /\ -Del

       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssue
       ONE OF DELETE FROM emplStatus[Employee*Status]
               SELECTFROM noNecessaryEqtHasBeenIssued;(-((emplStatus /\ -Delta);

              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
              DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM emplStatus;(-('Grey'[Status];(emplStatus /\ -Delta)~)

              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM allNecessaryEqtHasBeenIssued;(-((emplStatus /\ -Delta)

              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM emplStatus;(-('Grey'[Status];(emplStatus /\ -Delta)~)
              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM needsToReturnEqt;(-((emplStatus /\ -Delta);'Grey'[Stat

              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM emplStatus;(-('Grey'[Status];(emplStatus /\ -Delta)~)

              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessary

              (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       (MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;all
       ONE OF DELETE FROM emplStatus[Employee*Status]
               SELECTFROM noNecessaryEqtHasBeenIssued;(-(emplStatus /\ -Delta) /

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
              DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM emplStatus;'Grey'[Status];(-(emplStatus /\ -Delta)~ /\

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM allNecessaryEqtHasBeenIssued;(-(emplStatus /\ -Delta)

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM emplStatus;'Grey'[Status];(-(emplStatus /\ -Delta)~ /\

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM needsToReturnEqt;(-(emplStatus /\ -Delta) /\ noNecessa

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM emplStatus;'Grey'[Status];(-(emplStatus /\ -Delta)~ /\

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM (-(emplStatus /\ -Delta) /\ noNecessaryEqtHasBeenIssue

              (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       (MAINTAINING -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued /\
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessa

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM noNecessaryEqtHasBeenIssued~;(-((emplStatus /\ -Delta)
              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM (-((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessa

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM allNecessaryEqtHasBeenIssued~;(-((emplStatus /\ -Delta

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM (-((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessa

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM needsToReturnEqt~;(-((emplStatus /\ -Delta);'Grey'[Sta

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessary

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       (MAINTAINING -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryEqtHa
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM noNecessaryEqtHasBeenIssued~;((-emplStatus /\ noNecessa

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM allNecessaryEqtHasBeenIssued~;((-emplStatus /\ noNeces

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM needsToReturnEqt~;((-emplStatus /\ noNecessaryEqtHasBe

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
              DELETE FROM emplStatus[Employee*Status]
               SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
                   (MAINTAINING -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /\ a
                   ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                           SELECTFROM -((emplStatus /\ -Delta);'Blue'[Status];(emplStatus /\

                          (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
                          DELETE FROM needsToReturnEqt[Employee*Employee]
                           SELECTFROM -((emplStatus /\ -Delta);'Blue'[Status];(emplStatus /\

                          (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
                          DELETE FROM Isn{detyp=Employee}
                           SELECTFROM -((emplStatus /\ -Delta);'Blue'[Status];(emplStatus /\

                          (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
                   (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Empl
                   ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
                           SELECTFROM -((emplStatus /\ -Delta);'Orange'[Status];(emplStatus

                          (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Or
                          DELETE FROM Isn{detyp=Employee}
                           SELECTFROM -((emplStatus /\ -Delta);'Orange'[Status];(emplStatus

                          (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Or
                   (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[St
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ I[
            (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'[S
            (MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecess
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
            (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
            (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee])
            (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];e
```

------ Derivation ----->

```
ALL of ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);'Black'[Status];(emplStatus /\ -De

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -((emplStatus /\ -Delta);'Black'[Status];(emplStatus /\ -De

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -((emplStatus /\ -Delta);'Black'[Status];(emplStatus /\ -De
       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\
ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((emplStatus /\ -Delta);'Green'[Status];(emplStatus /\ -De

       (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -((emplStatus /\ -Delta);'Green'[Status];(emplStatus /\ -De

       (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Gre
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((emplStatus /\ -Delta);'Red'[Status];(emplStatus /\ -Delt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStat
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -((emplStatus /\ -Delta);'Red'[Status];(emplStatus /\ -Delt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStat
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'
DELETE FROM Isn{detyp=Employee}
 SELECTFROM -((emplStatus /\ -Delta);'Yellow'[Status];(emplStatus /\ -Delta)~)

(TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNec
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\ -Del

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\ -Del

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\ -Del

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status];(emplStatus /\ -Del

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;(-((emplStatus /\ -Delta);'Grey

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];(emplStatus /\ -Delta)~) /\ em

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM emplStatus[Employee*Status]
```

```
        SELECTFROM allNecessaryEqtHasBeenIssued;(-((emplStatus /\ -Delta);'Gre

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];(emplStatus /\ -Delta)~) /\ em

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;(-((emplStatus /\ -Delta);'Grey'[Status])

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];(emplStatus /\ -Delta)~) /\ em

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessaryEqtH

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;allNeces
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;(-(emplStatus /\ -Delta) /\ noN

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-(emplStatus /\ -Delta)~ /\ 'Gre

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;(-(emplStatus /\ -Delta) /\ no

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-(emplStatus /\ -Delta)~ /\ 'Gre

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;(-(emplStatus /\ -Delta) /\ noNecessaryEqt

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-(emplStatus /\ -Delta)~ /\ 'Gre

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM (-(emplStatus /\ -Delta) /\ noNecessaryEqtHasBeenIssued~;em

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued
(MAINTAINING -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued /\ 'Grey
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
```

```
        SELECTFROM (-((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessaryEqt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;(-((emplStatus /\ -Delta);'Gre

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM (-((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessaryEqt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;(-((emplStatus /\ -Delta);'Gr

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM (-((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessaryEqt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;(-((emplStatus /\ -Delta);'Grey'[Status])

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM -((emplStatus /\ -Delta);'Grey'[Status]) /\ noNecessaryEqtH

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
(MAINTAINING -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryEqtHasBee
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus;'Gr

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;((-emplStatus /\ noNecessaryEq

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus;'Gr

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;((-emplStatus /\ noNecessaryE

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus;'Gr

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;((-emplStatus /\ noNecessaryEqtHasBeenIss
                (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
                DELETE FROM emplStatus[Employee*Status]
                 SELECTFROM ((-emplStatus /\ noNecessaryEqtHasBeenIssued;emplStatus;'Gr

                (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /
         (MAINTAINING -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /\ allNec
         ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                 SELECTFROM -((emplStatus /\ -Delta);'Blue'[Status];(emplStatus /\ -Del

                (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
                DELETE FROM needsToReturnEqt[Employee*Employee]
                 SELECTFROM -((emplStatus /\ -Delta);'Blue'[Status];(emplStatus /\ -Del

                (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
                DELETE FROM Isn{detyp=Employee}
                 SELECTFROM -((emplStatus /\ -Delta);'Blue'[Status];(emplStatus /\ -Del

                (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
         (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee]
         ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
                 SELECTFROM -((emplStatus /\ -Delta);'Orange'[Status];(emplStatus /\ -D

                (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'
                DELETE FROM Isn{detyp=Employee}
                 SELECTFROM -((emplStatus /\ -Delta);'Orange'[Status];(emplStatus /\ -D

                (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'
         (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status]
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ I[Emplo
  (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green'[Sta
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'[Status
  (MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecessaryEq
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
  (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
  (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee]) \/ em
  (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];emplSt
```

<------End Derivation --

```
 ON INSERT Delta IN sessionEmployee[SESSION*Employee] EXECUTE    -- (ECA rule 55)
 ALL of INSERT INTO Isn{detyp=Employee}
         SELECTFROM ((sessionEmployee \/ Delta)~;sessionEmployee /\ -I[Employee])

        (TO MAINTAIN  -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI
        INSERT INTO Isn{detyp=SESSION}
         SELECTFROM (Delta;Delta~ /\ I[SESSION]) - I[SESSION]

 (MAINTAINING -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI session
```

------ Derivation ----->

```
ALL of INSERT INTO Isn{detyp=Employee}
        SELECTFROM ((sessionEmployee \/ Delta)~;sessionEmployee /\ -I[Employee]) \/ (

       (TO MAINTAIN  -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI sess
       INSERT INTO Isn{detyp=SESSION}
        SELECTFROM (Delta;Delta~ /\ I[SESSION]) - I[SESSION]

(MAINTAINING -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI sessionEmplo
```

<------End Derivation --

```
     ON INSERT Delta IN sessionOrgRole[SESSION*OrganizationalRole] EXECUTE    -- (ECA
     ALL of INSERT INTO Isn{detyp=OrganizationalRole}
             SELECTFROM ((sessionOrgRole \/ Delta)~;sessionOrgRole /\ -I[Organizationa

            (TO MAINTAIN  -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] I
            INSERT INTO Isn{detyp=SESSION}
             SELECTFROM (Delta;Delta~ /\ I[SESSION]) - I[SESSION]

     (MAINTAINING -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] FROM UNI
```

------ Derivation ----->

```
ALL of INSERT INTO Isn{detyp=OrganizationalRole}
        SELECTFROM ((sessionOrgRole \/ Delta)~;sessionOrgRole /\ -I[OrganizationalRol

       (TO MAINTAIN  -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] FROM
       INSERT INTO Isn{detyp=SESSION}
        SELECTFROM (Delta;Delta~ /\ I[SESSION]) - I[SESSION]

(MAINTAINING -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] FROM UNI sess
```

<------End Derivation --

```
     ON INSERT Delta IN Isn{detyp=Employee} EXECUTE    -- (ECA rule 59)
     BLOCK
     (CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
     (CANNOT CHANGE V[Employee*Employee] FROM IRF emplManager::Employee*Employee)
```

------ Derivation ----->

```
     BLOCK
     (CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
     (CANNOT CHANGE V[Employee*Employee] FROM IRF emplManager::Employee*Employee)
```

<------End Derivation --

```
          ON DELETE Delta FROM Isn{detyp=Employee} EXECUTE    -- (ECA rule 60)
          ONE OF DELETE FROM emplManager[Employee*Employee]
                  SELECTFROM -(emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organizat

                 (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Ma
                 DELETE FROM emplManager[Employee*Employee]
                  SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/ (-

                 (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;
                 DELETE FROM emplManager[Employee*Employee]
                  SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/ (-

                 (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;
                 DELETE FROM emplManager[Employee*Employee]
                  SELECTFROM maEmployee~;maManager;(-I[Employee] /\ maManager~;maEmployee;

                 (TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM M
                 DELETE FROM maEmployee[ManagerApproval*Employee]
                  SELECTFROM maManager;(-I[Employee] /\ maManager~;maEmployee;emplManager)

                 (TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM M
                 DELETE FROM maManager[ManagerApproval*Employee]
                  SELECTFROM maEmployee;emplManager;(-I[Employee] /\ emplManager~;maEmploy

                 (TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM M
                 DELETE FROM needsToReturnEqt[Employee*Employee]
                  SELECTFROM -I[Employee] /\ needsToReturnEqt

                 (TO MAINTAIN  -needsToReturnEqt \/ I[Employee] FROM delneedsToReturnEqt)
                 DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                  SELECTFROM -I[Employee] /\ allNecessaryEqtHasBeenIssued

                 (TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ I[Employee] FROM delallNec
                 DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
                  SELECTFROM -I[Employee] /\ noNecessaryEqtHasBeenIssued

                 (TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ I[Employee] FROM delnoNeces
                 DELETE FROM emplManager[Employee*Employee]
                  SELECTFROM -I[Employee] /\ emplManager /\ emplManager~
```

```
(TO MAINTAIN  -(emplManager /\ emplManager~) \/ I[Employee] FROM ASY empl
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM -I[Employee] /\ emplManager~ /\ emplManager

(TO MAINTAIN  -(emplManager /\ emplManager~) \/ I[Employee] FROM ASY empl
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM emplManager;(-I[Employee] /\ emplManager~;emplManager)

(TO MAINTAIN  -(emplManager~;emplManager) \/ I[Employee] FROM UNI emplMana
DELETE FROM emplIssuedEqt[Employee*Equipment]
 SELECTFROM (-I[Employee] /\ emplIssuedEqt;emplIssuedEqt~);emplIssuedEqt

(TO MAINTAIN  -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ empl
DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM (-I[Employee] /\ emplOwnsEqt;emplOwnsEqt~);emplOwnsEqt

(TO MAINTAIN  -(emplOwnsEqt;emplOwnsEqt~) \/ I[Employee] FROM INJ emplOwns
DELETE FROM maEmployee[ManagerApproval*Employee]
 SELECTFROM maEmployee;(-I[Employee] /\ maEmployee~;maEmployee)

(TO MAINTAIN  -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmploye
DELETE FROM maManager[ManagerApproval*Employee]
 SELECTFROM maManager;(-I[Employee] /\ maManager~;maManager)

(TO MAINTAIN  -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::
DELETE FROM sessionEmployee[SESSION*Employee]
 SELECTFROM sessionEmployee;(-I[Employee] /\ sessionEmployee~;sessionEmpl

(TO MAINTAIN  -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI
DELETE FROM emplName[Employee*EmployeeName]
 SELECTFROM Delta;V[Employee*EmployeeName]

DELETE FROM emplManager[Employee*Employee]
 SELECTFROM Delta;V[Employee*Employee]

DELETE FROM emplManager[Employee*Employee]
 SELECTFROM V[Employee*Employee];Delta

DELETE FROM emplOrgRole[Employee*OrganizationalRole]
 SELECTFROM Delta;V[Employee*OrganizationalRole]

DELETE FROM emplIssuedEqt[Employee*Equipment]
 SELECTFROM Delta;V[Employee*Equipment]

DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM Delta;V[Employee*Equipment]

DELETE FROM maEmployee[ManagerApproval*Employee]
 SELECTFROM V[ManagerApproval*Employee];Delta
```

```
                DELETE FROM maManager[ManagerApproval*Employee]
                 SELECTFROM V[ManagerApproval*Employee];Delta

                DELETE FROM needsToReturnEqt[Employee*Employee]
                 SELECTFROM Delta;V[Employee*Employee]

                DELETE FROM needsToReturnEqt[Employee*Employee]
                 SELECTFROM V[Employee*Employee];Delta

                DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                 SELECTFROM Delta;V[Employee*Employee]

                DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
                 SELECTFROM V[Employee*Employee];Delta

                DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
                 SELECTFROM Delta;V[Employee*Employee]

                DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
                 SELECTFROM V[Employee*Employee];Delta

                DELETE FROM emplStatus[Employee*Status]
                 SELECTFROM Delta;V[Employee*Status]

                DELETE FROM sessionEmployee[SESSION*Employee]
                 SELECTFROM V[SESSION*Employee];Delta

         (MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
         (MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
         (MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
         (MAINTAINING -needsToReturnEqt \/ I[Employee] FROM delneedsToReturnEqt)
         (MAINTAINING -allNecessaryEqtHasBeenIssued \/ I[Employee] FROM delallNecessaryEq
         (MAINTAINING -noNecessaryEqtHasBeenIssued \/ I[Employee] FROM delnoNecessaryEqtH
         (MAINTAINING -(emplManager /\ emplManager~) \/ I[Employee] FROM ASY emplManager:
         (MAINTAINING -(emplManager~;emplManager) \/ I[Employee] FROM UNI emplManager::Em
         (MAINTAINING -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ emplIssuedE
         (MAINTAINING -(emplOwnsEqt;emplOwnsEqt~) \/ I[Employee] FROM INJ emplOwnsEqt::Em
         (MAINTAINING -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::Manag
         (MAINTAINING -I[ManagerApproval] \/ maEmployee;maEmployee~ FROM TOT maEmployee::
         (MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerA
         (MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::Mana
         (MAINTAINING -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI session
```

------ Derivation ----->

```
     ONE OF DELETE FROM emplManager[Employee*Employee]
             SELECTFROM -(emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organizational
(TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;’Manager
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/ (-(empl

(TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;’Mana
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/ (-(empl

(TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;’Mana
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM maEmployee~;maManager;(-I[Employee] /\ maManager~;maEmployee;emplM

(TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM Manage
DELETE FROM maEmployee[ManagerApproval*Employee]
 SELECTFROM maManager;(-I[Employee] /\ maManager~;maEmployee;emplManager);empl

(TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM Manage
DELETE FROM maManager[ManagerApproval*Employee]
 SELECTFROM maEmployee;emplManager;(-I[Employee] /\ emplManager~;maEmployee~;m

(TO MAINTAIN  -(emplManager~;maEmployee~;maManager) \/ I[Employee] FROM Manage
DELETE FROM needsToReturnEqt[Employee*Employee]
 SELECTFROM -I[Employee] /\ needsToReturnEqt

(TO MAINTAIN  -needsToReturnEqt \/ I[Employee] FROM delneedsToReturnEqt)
DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
 SELECTFROM -I[Employee] /\ allNecessaryEqtHasBeenIssued

(TO MAINTAIN  -allNecessaryEqtHasBeenIssued \/ I[Employee] FROM delallNecessar
DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
 SELECTFROM -I[Employee] /\ noNecessaryEqtHasBeenIssued

(TO MAINTAIN  -noNecessaryEqtHasBeenIssued \/ I[Employee] FROM delnoNecessaryE
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM -I[Employee] /\ emplManager /\ emplManager~

(TO MAINTAIN  -(emplManager /\ emplManager~) \/ I[Employee] FROM ASY emplManag
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM -I[Employee] /\ emplManager~ /\ emplManager

(TO MAINTAIN  -(emplManager /\ emplManager~) \/ I[Employee] FROM ASY emplManag
DELETE FROM emplManager[Employee*Employee]
 SELECTFROM emplManager;(-I[Employee] /\ emplManager~;emplManager)

(TO MAINTAIN  -(emplManager~;emplManager) \/ I[Employee] FROM UNI emplManager:
DELETE FROM emplIssuedEqt[Employee*Equipment]
 SELECTFROM (-I[Employee] /\ emplIssuedEqt;emplIssuedEqt~);emplIssuedEqt

(TO MAINTAIN  -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ emplIssu
DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM (-I[Employee] /\ emplOwnsEqt;emplOwnsEqt~);emplOwnsEqt

(TO MAINTAIN  -(emplOwnsEqt;emplOwnsEqt~) \/ I[Employee] FROM INJ emplOwnsEqt:
DELETE FROM maEmployee[ManagerApproval*Employee]
 SELECTFROM maEmployee;(-I[Employee] /\ maEmployee~;maEmployee)

(TO MAINTAIN  -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::Ma
DELETE FROM maManager[ManagerApproval*Employee]
 SELECTFROM maManager;(-I[Employee] /\ maManager~;maManager)

(TO MAINTAIN  -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::Manag
DELETE FROM sessionEmployee[SESSION*Employee]
 SELECTFROM sessionEmployee;(-I[Employee] /\ sessionEmployee~;sessionEmployee)

(TO MAINTAIN  -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI sess
DELETE FROM emplName[Employee*EmployeeName]
 SELECTFROM Delta;V[Employee*EmployeeName]

DELETE FROM emplManager[Employee*Employee]
 SELECTFROM Delta;V[Employee*Employee]

DELETE FROM emplManager[Employee*Employee]
 SELECTFROM V[Employee*Employee];Delta

DELETE FROM emplOrgRole[Employee*OrganizationalRole]
 SELECTFROM Delta;V[Employee*OrganizationalRole]

DELETE FROM emplIssuedEqt[Employee*Equipment]
 SELECTFROM Delta;V[Employee*Equipment]

DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM Delta;V[Employee*Equipment]

DELETE FROM maEmployee[ManagerApproval*Employee]
 SELECTFROM V[ManagerApproval*Employee];Delta

DELETE FROM maManager[ManagerApproval*Employee]
 SELECTFROM V[ManagerApproval*Employee];Delta

DELETE FROM needsToReturnEqt[Employee*Employee]
 SELECTFROM Delta;V[Employee*Employee]

DELETE FROM needsToReturnEqt[Employee*Employee]
 SELECTFROM V[Employee*Employee];Delta

DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
 SELECTFROM Delta;V[Employee*Employee]

DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
```

```
          SELECTFROM V[Employee*Employee];Delta

         DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
          SELECTFROM Delta;V[Employee*Employee]

         DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
          SELECTFROM V[Employee*Employee];Delta

         DELETE FROM emplStatus[Employee*Status]
          SELECTFROM Delta;V[Employee*Status]

         DELETE FROM sessionEmployee[SESSION*Employee]
          SELECTFROM V[SESSION*Employee];Delta

  (MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
  (MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
  (MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
  (MAINTAINING -needsToReturnEqt \/ I[Employee] FROM delneedsToReturnEqt)
  (MAINTAINING -allNecessaryEqtHasBeenIssued \/ I[Employee] FROM delallNecessaryEqtHasB
  (MAINTAINING -noNecessaryEqtHasBeenIssued \/ I[Employee] FROM delnoNecessaryEqtHasBee
  (MAINTAINING -(emplManager /\ emplManager~) \/ I[Employee] FROM ASY emplManager::Empl
  (MAINTAINING -(emplManager~;emplManager) \/ I[Employee] FROM UNI emplManager::Employe
  (MAINTAINING -(emplIssuedEqt;emplIssuedEqt~) \/ I[Employee] FROM INJ emplIssuedEqt::E
  (MAINTAINING -(emplOwnsEqt;emplOwnsEqt~) \/ I[Employee] FROM INJ emplOwnsEqt::Employe
  (MAINTAINING -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::ManagerApp
  (MAINTAINING -I[ManagerApproval] \/ maEmployee;maEmployee~ FROM TOT maEmployee::Manag
  (MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerApprov
  (MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::ManagerA
  (MAINTAINING -(sessionEmployee~;sessionEmployee) \/ I[Employee] FROM UNI sessionEmplo
```

<------End Derivation --

```
     ON INSERT Delta IN Isn{detyp=OrganizationalRole} EXECUTE    -- (ECA rule 61)
     BLOCK
     (CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Employee*Employee] FROM Directors do not have a manager)
```

<------End Derivation --

```
     ON DELETE Delta FROM Isn{detyp=OrganizationalRole} EXECUTE    -- (ECA rule 62)
     ALL of DELETE FROM emplManager[Employee*Employee]
```

```
             SELECTFROM -(emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organizat

            (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Ma
            DELETE FROM Isn{detyp=Employee}
             SELECTFROM (-(emplManager;emplManager~) /\ -(emplOrgRole;'Director'[Orga

            (TO MAINTAIN  -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Di
            (TO MAINTAIN  -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];
            DELETE FROM sessionOrgRole[SESSION*OrganizationalRole]
             SELECTFROM sessionOrgRole;(-I[OrganizationalRole] /\ sessionOrgRole~;ses

            (TO MAINTAIN  -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] 
            DELETE FROM emplOrgRole[Employee*OrganizationalRole]
             SELECTFROM V[Employee*OrganizationalRole];Delta

            DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
             SELECTFROM Delta;V[OrganizationalRole*EqtKind]

            ONE OF DELETE FROM emplManager[Employee*Employee]
                    SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager

                   (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplO
                   DELETE FROM emplManager[Employee*Employee]
                    SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager

                   (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplO
            (MAINTAINING -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;'
     (MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
     (MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[O
     (MAINTAINING -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Director'[
     (MAINTAINING -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];emplOrgR
     (MAINTAINING -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] FROM UNI
```

------ Derivation ----->

```
ALL of DELETE FROM emplManager[Employee*Employee]
        SELECTFROM -(emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organizational

       (TO MAINTAIN  -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM (-(emplManager;emplManager~) /\ -(emplOrgRole;'Director'[Organizat

       (TO MAINTAIN  -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Directo
       (TO MAINTAIN  -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];emplO
       DELETE FROM sessionOrgRole[SESSION*OrganizationalRole]
        SELECTFROM sessionOrgRole;(-I[OrganizationalRole] /\ sessionOrgRole~;sessionO

       (TO MAINTAIN  -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] FROM 
       DELETE FROM emplOrgRole[Employee*OrganizationalRole]
        SELECTFROM V[Employee*OrganizationalRole];Delta

       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM Delta;V[OrganizationalRole*EqtKind]

       ONE OF DELETE FROM emplManager[Employee*Employee]
               SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/

              (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRol
              DELETE FROM emplManager[Employee*Employee]
               SELECTFROM emplManager;((-I[Employee] /\ emplManager~;emplManager) \/

              (TO MAINTAIN  -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRol
       (MAINTAINING -(emplManager~;emplManager) \/ (I[Employee] /\ emplOrgRole;'Manag
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
(MAINTAINING -emplManager \/ emplManager;(I[Employee] /\ emplOrgRole;'Manager'[Organi
(MAINTAINING -I[Employee] \/ emplManager;emplManager~ \/ emplOrgRole;'Director'[Organ
(MAINTAINING -I[Employee] \/ emplOrgRole;'Employee'[OrganizationalRole];emplOrgRole~
(MAINTAINING -(sessionOrgRole~;sessionOrgRole) \/ I[OrganizationalRole] FROM UNI sess
```

<------End Derivation --

```
ON INSERT Delta IN Isn{detyp=Equipment} EXECUTE    -- (ECA rule 63)
BLOCK
(CANNOT CHANGE V[Equipment*Equipment] FROM Coherence of registered equipment)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE V[Equipment*Equipment] FROM Coherence of registered equipment)
```

<------End Derivation --

```
ON DELETE Delta FROM Isn{detyp=Equipment} EXECUTE    -- (ECA rule 64)
ONE OF DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM (-I[Equipment] /\ eqtApprovedProp;emplIssuedEqt~;emplIssuedEq

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtAppro
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;(-I[Equipment] /\ emplIssuedEqt~;emplIssuedEqt;

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtAppro
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedP
       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtAppro
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ eqtApprovedProp;emplIssuedEqt~;emplIssuedEqt

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtAppro
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;eqtApprovedProp;(-I[Equipment] /\ eqtApprovedPr

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipm
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;(-I[Equipment] /\ emplIssuedEqt~;emplIssuedEqt;

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipm
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM emplIssuedEqt~;emplIssuedEqt;(-I[Equipment] /\ emplIssuedEqt~

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipm
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipm
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM (-I[Equipment] /\ eqtApprovedProp;emplOwnsEqt~;emplOwnsEqt /\

       (TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedP
       DELETE FROM emplOwnsEqt[Employee*Equipment]
        SELECTFROM emplOwnsEqt;(-I[Equipment] /\ emplOwnsEqt~;emplOwnsEqt;eqtAppr

       (TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedP
       DELETE FROM emplOwnsEqt[Employee*Equipment]
        SELECTFROM emplOwnsEqt;eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedPro

       (TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedP
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ eqtApprovedProp;emplOwnsEqt~;emplOwnsEqt /\

       (TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedP
       DELETE FROM emplOwnsEqt[Employee*Equipment]
        SELECTFROM emplOwnsEqt;eqtApprovedProp;(-I[Equipment] /\ eqtApprovedProp

       (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment]
       DELETE FROM emplOwnsEqt[Employee*Equipment]
        SELECTFROM emplOwnsEqt;(-I[Equipment] /\ emplOwnsEqt~;emplOwnsEqt;eqtAppr

       (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment]
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM emplOwnsEqt~;emplOwnsEqt;(-I[Equipment] /\ emplOwnsEqt~;emplO

       (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment]
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp /\

       (TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment]
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -(emplIssuedEqt~

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKin
       DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
        SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -(emplIssuedEqt~

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKin
       DELETE FROM Isn{detyp=EqtKind}
        SELECTFROM -(eqtKind~;(I[Equipment] /\ -(emplIssuedEqt~;emplIssuedEqt));

       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKin
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM (-I[Equipment] /\ eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Y

       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer]
       DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
        SELECTFROM eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedProp;eqtApprove

       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer]
       DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
        SELECTFROM (-I[Equipment] /\ eqtApprovedBySecOff;'Yes'[Yes/No answer];eq

       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer]
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Ye

       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer]
       DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
        SELECTFROM (-I[Equipment] /\ eqtApprovedBySecOff;'Yes'[Yes/No answer];eq

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
       DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
        SELECTFROM eqtApprovedProp;(-I[Equipment] /\ eqtApprovedProp~;eqtApprove

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ eqtApprovedBySecOff;'Yes'[Yes/No answer];eqt

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ eqtApprovedProp
```

```
        (TO MAINTAIN  -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
        DELETE FROM eqtApprovedProp[Equipment*Equipment]
         SELECTFROM (-I[Equipment] /\ eqtApprovedProp;eqtApprovedProp);eqtApprove

        (TO MAINTAIN  -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI
        DELETE FROM eqtApprovedProp[Equipment*Equipment]
         SELECTFROM eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedProp;eqtApprove

        (TO MAINTAIN  -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI
        DELETE FROM emplIssuedEqt[Employee*Equipment]
         SELECTFROM V[Employee*Equipment];Delta

        DELETE FROM emplOwnsEqt[Employee*Equipment]
         SELECTFROM V[Employee*Equipment];Delta

        DELETE FROM eqtMake[Equipment*EqtMake]
         SELECTFROM Delta;V[Equipment*EqtMake]

        DELETE FROM eqtType[Equipment*EqtType]
         SELECTFROM Delta;V[Equipment*EqtType]

        DELETE FROM eqtSerial[Equipment*EqtSerial]
         SELECTFROM Delta;V[Equipment*EqtSerial]

        DELETE FROM eqtKind[Equipment*EqtKind]
         SELECTFROM Delta;V[Equipment*EqtKind]

        DELETE FROM eqtStatus[Equipment*EqtStatus]
         SELECTFROM Delta;V[Equipment*EqtStatus]

        DELETE FROM eqtID[Equipment*EqtCompanyID]
         SELECTFROM Delta;V[Equipment*EqtCompanyID]

        DELETE FROM eqtApprovedProp[Equipment*Equipment]
         SELECTFROM Delta;V[Equipment*Equipment]

        DELETE FROM eqtApprovedProp[Equipment*Equipment]
         SELECTFROM V[Equipment*Equipment];Delta

        DELETE FROM eqtSecReqt[Equipment*SecRequirement]
         SELECTFROM Delta;V[Equipment*SecRequirement]

        DELETE FROM eqtSatReqt[Equipment*SecRequirement]
         SELECTFROM Delta;V[Equipment*SecRequirement]

        DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
         SELECTFROM Delta;V[Equipment*Yes/No answer]

(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp I
          (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp
          (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
          (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
          (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Eq
          (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\
          (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\
          (MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
          (MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM ASY eqtApprovedProp::Equipmen
          (MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtAppro
          (MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM INJ eqtAppro
```

------ Derivation ----->

```
ONE OF DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM (-I[Equipment] /\ eqtApprovedProp;emplIssuedEqt~;emplIssuedEqt /\

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtApprovedPr
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;(-I[Equipment] /\ emplIssuedEqt~;emplIssuedEqt;eqtAp

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtApprovedPr
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedProp;e

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtApprovedPr
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ eqtApprovedProp;emplIssuedEqt~;emplIssuedEqt /\ e

       (TO MAINTAIN  -(eqtApprovedProp~;emplIssuedEqt~;emplIssuedEqt /\ eqtApprovedPr
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;eqtApprovedProp;(-I[Equipment] /\ eqtApprovedProp~;e

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipment];
       DELETE FROM emplIssuedEqt[Employee*Equipment]
        SELECTFROM emplIssuedEqt;(-I[Equipment] /\ emplIssuedEqt~;emplIssuedEqt;eqtAp

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipment];
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM emplIssuedEqt~;emplIssuedEqt;(-I[Equipment] /\ emplIssuedEqt~;empl

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipment];
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -I[Equipment] /\ emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp /\ e

       (TO MAINTAIN  -(emplIssuedEqt~;emplIssuedEqt;eqtApprovedProp~ /\ I[Equipment];
       DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM (-I[Equipment] /\ eqtApprovedProp;emplOwnsEqt~;emplOwnsEqt /\ eqtA
(TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedProp~;
DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM emplOwnsEqt;(-I[Equipment] /\ emplOwnsEqt~;emplOwnsEqt;eqtApproved

(TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedProp~;
DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM emplOwnsEqt;eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedProp;emp

(TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedProp~;
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM -I[Equipment] /\ eqtApprovedProp;emplOwnsEqt~;emplOwnsEqt /\ eqtAp

(TO MAINTAIN  -(eqtApprovedProp~;emplOwnsEqt~;emplOwnsEqt /\ eqtApprovedProp~;
DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM emplOwnsEqt;eqtApprovedProp;(-I[Equipment] /\ eqtApprovedProp~;emp

(TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment];eqtA
DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM emplOwnsEqt;(-I[Equipment] /\ emplOwnsEqt~;emplOwnsEqt;eqtApproved

(TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment];eqtA
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM emplOwnsEqt~;emplOwnsEqt;(-I[Equipment] /\ emplOwnsEqt~;emplOwnsEq

(TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment];eqtA
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM -I[Equipment] /\ emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp /\ eqtAp

(TO MAINTAIN  -(emplOwnsEqt~;emplOwnsEqt;eqtApprovedProp~ /\ I[Equipment];eqtA
DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
 SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -(emplIssuedEqt~;empl

(TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I
DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
 SELECTFROM stdIssueEqtKind;(-(eqtKind~;(I[Equipment] /\ -(emplIssuedEqt~;empl

(TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I
DELETE FROM Isn{detyp=EqtKind}
 SELECTFROM -(eqtKind~;(I[Equipment] /\ -(emplIssuedEqt~;emplIssuedEqt));eqtKi

(TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM (-I[Equipment] /\ eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Yes/No

(TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtA
DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
 SELECTFROM eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedProp;eqtApprovedBySe

(TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtA
DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
```

```
 SELECTFROM (-I[Equipment] /\ eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtAppr

(TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtA
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM -I[Equipment] /\ eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Yes/No

(TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtA
DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
 SELECTFROM (-I[Equipment] /\ eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtAppr

(TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;
DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
 SELECTFROM eqtApprovedProp;(-I[Equipment] /\ eqtApprovedProp~;eqtApprovedBySe

(TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;(-I[

(TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM -I[Equipment] /\ eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtAppro

(TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM -I[Equipment] /\ eqtApprovedProp

(TO MAINTAIN  -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM (-I[Equipment] /\ eqtApprovedProp;eqtApprovedProp);eqtApprovedProp

(TO MAINTAIN  -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtA
DELETE FROM eqtApprovedProp[Equipment*Equipment]
 SELECTFROM eqtApprovedProp~;(-I[Equipment] /\ eqtApprovedProp;eqtApprovedProp

(TO MAINTAIN  -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtA
DELETE FROM emplIssuedEqt[Employee*Equipment]
 SELECTFROM V[Employee*Equipment];Delta

DELETE FROM emplOwnsEqt[Employee*Equipment]
 SELECTFROM V[Employee*Equipment];Delta

DELETE FROM eqtMake[Equipment*EqtMake]
 SELECTFROM Delta;V[Equipment*EqtMake]

DELETE FROM eqtType[Equipment*EqtType]
 SELECTFROM Delta;V[Equipment*EqtType]

DELETE FROM eqtSerial[Equipment*EqtSerial]
 SELECTFROM Delta;V[Equipment*EqtSerial]
```

```
            DELETE FROM eqtKind[Equipment*EqtKind]
             SELECTFROM Delta;V[Equipment*EqtKind]

            DELETE FROM eqtStatus[Equipment*EqtStatus]
             SELECTFROM Delta;V[Equipment*EqtStatus]

            DELETE FROM eqtID[Equipment*EqtCompanyID]
             SELECTFROM Delta;V[Equipment*EqtCompanyID]

            DELETE FROM eqtApprovedProp[Equipment*Equipment]
             SELECTFROM Delta;V[Equipment*Equipment]

            DELETE FROM eqtApprovedProp[Equipment*Equipment]
             SELECTFROM V[Equipment*Equipment];Delta

            DELETE FROM eqtSecReqt[Equipment*SecRequirement]
             SELECTFROM Delta;V[Equipment*SecRequirement]

            DELETE FROM eqtSatReqt[Equipment*SecRequirement]
             SELECTFROM Delta;V[Equipment*SecRequirement]

            DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
             SELECTFROM Delta;V[Equipment*Yes/No answer]

     (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
     (MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtApprovedProp FROM
     (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM Equi
     (MAINTAINING -(emplOwnsEqt~;emplOwnsEqt /\ I[Equipment]) \/ eqtApprovedProp FROM Equi
     (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Equipme
     (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
     (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
     (MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM deleqtApprovedProp)
     (MAINTAINING -eqtApprovedProp \/ I[Equipment] FROM ASY eqtApprovedProp::Equipment*Equ
     (MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM UNI eqtApprovedP
     (MAINTAINING -(eqtApprovedProp;eqtApprovedProp) \/ I[Equipment] FROM INJ eqtApprovedP
```

<------End Derivation --

```
          ON DELETE Delta FROM Isn{detyp=EqtCompanyID} EXECUTE    -- (ECA rule 66)
          ALL of DELETE FROM eqtID[Equipment*EqtCompanyID]
                  SELECTFROM eqtID;(-I[EqtCompanyID] /\ eqtID~;eqtID) \/ V[Equipment*EqtCom

                 (TO MAINTAIN  -(eqtID~;eqtID) \/ I[EqtCompanyID] FROM UNI eqtID::Equipmen
                 ONE OF DELETE FROM eqtID[Equipment*EqtCompanyID]
                         SELECTFROM emplIssuedEqt~;emplIssuedEqt;eqtID;(-I[EqtCompanyID] /

                        (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID
                        DELETE FROM emplIssuedEqt[Employee*Equipment]
```

```
               SELECTFROM emplIssuedEqt;eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIs

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID
              DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIs

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID
              DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM emplIssuedEqt~;emplIssuedEqt;eqtID;(-I[EqtCompanyID] /

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID
              DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIssuedEqt~;emplIs

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID
              DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIssuedEqt~;emplIs

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID
       (MAINTAINING -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[Equi
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FRO
(MAINTAINING -(eqtID~;eqtID) \/ I[EqtCompanyID] FROM UNI eqtID::Equipment*EqtCom
```

------ Derivation ----->

```
ALL of DELETE FROM eqtID[Equipment*EqtCompanyID]
        SELECTFROM eqtID;(-I[EqtCompanyID] /\ eqtID~;eqtID) \/ V[Equipment*EqtCompany

       (TO MAINTAIN  -(eqtID~;eqtID) \/ I[EqtCompanyID] FROM UNI eqtID::Equipment*Eqt
       ONE OF DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM emplIssuedEqt~;emplIssuedEqt;eqtID;(-I[EqtCompanyID] /\ eqt

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[E
              DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIssuedE

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[E
              DELETE FROM emplIssuedEqt[Employee*Equipment]
               SELECTFROM emplIssuedEqt;eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIssuedE

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[E
              DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM emplIssuedEqt~;emplIssuedEqt;eqtID;(-I[EqtCompanyID] /\ eqt

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[E
              DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIssuedEqt~;emplIssued
              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[E
              DELETE FROM eqtID[Equipment*EqtCompanyID]
               SELECTFROM eqtID;(-I[EqtCompanyID] /\ eqtID~;emplIssuedEqt~;emplIssued

              (TO MAINTAIN  -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[E
       (MAINTAINING -(eqtID~;emplIssuedEqt~;emplIssuedEqt;eqtID /\ eqtID~;I[Equipment
(MAINTAINING -(emplIssuedEqt~;emplIssuedEqt /\ I[Equipment]) \/ eqtID;eqtID~ FROM Iss
(MAINTAINING -(eqtID~;eqtID) \/ I[EqtCompanyID] FROM UNI eqtID::Equipment*EqtCompanyI
```

<------End Derivation --

```
ON DELETE Delta FROM Isn{detyp=EmployeeName} EXECUTE    -- (ECA rule 68)
ONE OF DELETE FROM emplName[Employee*EmployeeName]
        SELECTFROM emplName;(-I[EmployeeName] /\ emplName~;emplName)

       (TO MAINTAIN  -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName:
       DELETE FROM emplName[Employee*EmployeeName]
        SELECTFROM V[Employee*EmployeeName];Delta

(MAINTAINING -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Employe
(MAINTAINING -I[Employee] \/ emplName;emplName~ FROM TOT emplName::Employee*Empl
```

------ Derivation ----->

```
ONE OF DELETE FROM emplName[Employee*EmployeeName]
        SELECTFROM emplName;(-I[EmployeeName] /\ emplName~;emplName)

       (TO MAINTAIN  -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Empl
       DELETE FROM emplName[Employee*EmployeeName]
        SELECTFROM V[Employee*EmployeeName];Delta

(MAINTAINING -(emplName~;emplName) \/ I[EmployeeName] FROM UNI emplName::Employee*Emp
(MAINTAINING -I[Employee] \/ emplName;emplName~ FROM TOT emplName::Employee*EmployeeN
```

<------End Derivation --

```
ON INSERT Delta IN Isn{detyp=EqtKind} EXECUTE    -- (ECA rule 69)
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (stdIssueEqtKind~;stdIssueEqtKi
              THEN INSERT INTO eqtKind[Equipment*EqtKind]
                    SELECTFROM ’b’[Equipment]*’a’[EqtKind]

                   (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind
              PICK a,b FROM eqtKind;(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKi
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (’a’[Equipme
                                 THEN ALL of INSERT INTO Isn{detyp=Equipment}
```

```
                                       SELECTFROM 'a'[Equipment]*'b'[Equi

                                      (TO MAINTAIN  -(stdIssueEqtKind~;st
                                      ONE OF DELETE FROM emplIssuedEqt[Em
                                              SELECTFROM emplIssuedEqt;('

                                             (TO MAINTAIN  -(stdIssueEqtK
                                             DELETE FROM emplIssuedEqt[Em
                                              SELECTFROM emplIssuedEqt;('

                                             (TO MAINTAIN  -(stdIssueEqtK
                                      (MAINTAINING -(stdIssueEqtKind~;std
                               (MAINTAINING -(stdIssueEqtKind~;stdIssueEq
                          PICK a,b FROM (I[Equipment] /\ -(emplIssuedEqt~
                          THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                SELECTFROM 'a'[Equipment]*'b'[EqtKind]

                               (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueE
                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[E
                   NEW x:Equipment;
                     ALL of ALL of INSERT INTO Isn{detyp=Equipment}
                                    SELECTFROM 'a'[Equipment]*'b'[EqtKind

                                   (TO MAINTAIN  -(stdIssueEqtKind~;stdIs
                                   ONE OF DELETE FROM emplIssuedEqt[Emplo
                                           SELECTFROM emplIssuedEqt;('x'[

                                          (TO MAINTAIN  -(stdIssueEqtKind
                                          DELETE FROM emplIssuedEqt[Emplo
                                           SELECTFROM emplIssuedEqt;('a'[

                                          (TO MAINTAIN  -(stdIssueEqtKind
                                   (MAINTAINING -(stdIssueEqtKind~;stdIss
                            (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKi
                            INSERT INTO eqtKind[Equipment*EqtKind]
                             SELECTFROM 'x'[Equipment]*'a'[Equipment]*'b'

                            (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtK
                     (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I
                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[E
            (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind
NEW x:Equipment;
  ALL of INSERT INTO eqtKind[Equipment*EqtKind]
          SELECTFROM 'x'[Equipment]*(stdIssueEqtKind~;stdIssueEqtKind /\ I

         (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind])
         ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Equipment]
                       THEN ALL of INSERT INTO Isn{detyp=Equipment}
                                    SELECTFROM 'a'[Equipment]*'b'[Equipme
```

```
                                          (TO MAINTAIN  -(stdIssueEqtKind~;stdIs
                                          ONE OF DELETE FROM emplIssuedEqt[Emplo
                                                  SELECTFROM emplIssuedEqt;('b'[

                                                 (TO MAINTAIN  -(stdIssueEqtKind
                                                 DELETE FROM emplIssuedEqt[Emplo
                                                  SELECTFROM emplIssuedEqt;('a'[

                                                 (TO MAINTAIN  -(stdIssueEqtKind
                                          (MAINTAINING -(stdIssueEqtKind~;stdIss
                                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKi
                              PICK a,b FROM (I[Equipment] /\ -(emplIssuedEqt~;em
                              THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM 'a'[Equipment]*'b'[EqtKind]

                                   (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtK
                       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtK
                       NEW x:Equipment;
                         ALL of INSERT INTO Isn{detyp=Equipment}
                                 SELECTFROM 'x'[Equipment]*(stdIssueEqtKind~;std

                                (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind
                                ONE OF DELETE FROM emplIssuedEqt[Employee*Equipme
                                        SELECTFROM emplIssuedEqt;('x'[Equipment]

                                       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssue
                                       DELETE FROM emplIssuedEqt[Employee*Equipme
                                        SELECTFROM emplIssuedEqt;('x'[Equipment]

                                       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssue
                                (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind
                                INSERT INTO eqtKind[Equipment*EqtKind]
                                 SELECTFROM 'x'[Equipment]*'x'[Equipment]*(stdIs

                                (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind
                        (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[Eq
                       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtK
                (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \
         (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKi
       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Eq
```

------ Derivation ----->

```
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (stdIssueEqtKind~;stdIssueEqtKind /\
              THEN INSERT INTO eqtKind[Equipment*EqtKind]
                    SELECTFROM 'b'[Equipment]*'a'[EqtKind]
```

```
            (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/
       PICK a,b FROM eqtKind;(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind] /
       THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[Equipment]*'
                          THEN ALL of INSERT INTO Isn{detyp=Equipment}
                                       SELECTFROM 'a'[Equipment]*'b'[Equipment

                                      (TO MAINTAIN  -(stdIssueEqtKind~;stdIssu
                                      ONE OF DELETE FROM emplIssuedEqt[Employe
                                              SELECTFROM emplIssuedEqt;('b'[Eq

                                             (TO MAINTAIN  -(stdIssueEqtKind~;
                                             DELETE FROM emplIssuedEqt[Employe
                                              SELECTFROM emplIssuedEqt;('a'[Eq

                                             (TO MAINTAIN  -(stdIssueEqtKind~;
                                      (MAINTAINING -(stdIssueEqtKind~;stdIssue
                               (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind
                          PICK a,b FROM (I[Equipment] /\ -(emplIssuedEqt~;empl
                          THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                SELECTFROM 'a'[Equipment]*'b'[EqtKind]

                               (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKin
                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKin
                   NEW x:Equipment;
                     ALL of ALL of INSERT INTO Isn{detyp=Equipment}
                                    SELECTFROM 'a'[Equipment]*'b'[EqtKind]*'x'

                                   (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEq
                                   ONE OF DELETE FROM emplIssuedEqt[Employee*E
                                           SELECTFROM emplIssuedEqt;('x'[Equip

                                          (TO MAINTAIN  -(stdIssueEqtKind~;std
                                          DELETE FROM emplIssuedEqt[Employee*E
                                           SELECTFROM emplIssuedEqt;('a'[Equip

                                          (TO MAINTAIN  -(stdIssueEqtKind~;std
                                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqt
                            (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\
                            INSERT INTO eqtKind[Equipment*EqtKind]
                             SELECTFROM 'x'[Equipment]*'a'[Equipment]*'b'[EqtK

                            (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /
                     (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtK
                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKin
            (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[
NEW x:Equipment;
  ALL of INSERT INTO eqtKind[Equipment*EqtKind]
          SELECTFROM 'x'[Equipment]*(stdIssueEqtKind~;stdIssueEqtKind /\ I[Eqt
                (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eq
                ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Equipment]*(std
                              THEN ALL of INSERT INTO Isn{detyp=Equipment}
                                           SELECTFROM 'a'[Equipment]*'b'[Equipment]

                                          (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEq
                                          ONE OF DELETE FROM emplIssuedEqt[Employee*E
                                                  SELECTFROM emplIssuedEqt;('b'[Equip

                                                 (TO MAINTAIN  -(stdIssueEqtKind~;std
                                                 DELETE FROM emplIssuedEqt[Employee*E
                                                  SELECTFROM emplIssuedEqt;('a'[Equip

                                                 (TO MAINTAIN  -(stdIssueEqtKind~;std
                                          (MAINTAINING -(stdIssueEqtKind~;stdIssueEqt
                                   (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\
                              PICK a,b FROM (I[Equipment] /\ -(emplIssuedEqt~;emplIss
                              THEN INSERT INTO eqtKind[Equipment*EqtKind]
                                    SELECTFROM 'a'[Equipment]*'b'[EqtKind]

                                   (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /
                       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind])
                       NEW x:Equipment;
                         ALL of INSERT INTO Isn{detyp=Equipment}
                                 SELECTFROM 'x'[Equipment]*(stdIssueEqtKind~;stdIssue

                                (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I
                                ONE OF DELETE FROM emplIssuedEqt[Employee*Equipment]
                                        SELECTFROM emplIssuedEqt;('x'[Equipment]*(std

                                       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKi
                                       DELETE FROM emplIssuedEqt[Employee*Equipment]
                                        SELECTFROM emplIssuedEqt;('x'[Equipment]*(std

                                       (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKi
                                (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[
                                INSERT INTO eqtKind[Equipment*EqtKind]
                                 SELECTFROM 'x'[Equipment]*'x'[Equipment]*(stdIssueEq

                                (TO MAINTAIN  -(stdIssueEqtKind~;stdIssueEqtKind /\ I
                         (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind
                       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind])
                (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqt
         (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(
       (MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[
(MAINTAINING -(stdIssueEqtKind~;stdIssueEqtKind /\ I[EqtKind]) \/ eqtKind~;(I[Equipme
```

<------End Derivation --

```
    ON DELETE Delta FROM Isn{detyp=EqtKind} EXECUTE    -- (ECA rule 70)
    ONE OF DELETE FROM eqtKind[Equipment*EqtKind]
             SELECTFROM eqtKind;(-I[EqtKind] /\ eqtKind~;eqtKind)

            (TO MAINTAIN  -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipme
            DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
             SELECTFROM V[OrganizationalRole*EqtKind];Delta

            DELETE FROM eqtKind[Equipment*EqtKind]
             SELECTFROM V[Equipment*EqtKind];Delta

            DELETE FROM maEqtKind[ManagerApproval*EqtKind]
             SELECTFROM V[ManagerApproval*EqtKind];Delta

    (MAINTAINING -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*EqtKi
    (MAINTAINING -I[Equipment] \/ eqtKind;eqtKind~ FROM TOT eqtKind::Equipment*EqtKi
```

------ Derivation ----->

```
    ONE OF DELETE FROM eqtKind[Equipment*EqtKind]
             SELECTFROM eqtKind;(-I[EqtKind] /\ eqtKind~;eqtKind)

            (TO MAINTAIN  -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*Eq
            DELETE FROM stdIssueEqtKind[OrganizationalRole*EqtKind]
             SELECTFROM V[OrganizationalRole*EqtKind];Delta

            DELETE FROM eqtKind[Equipment*EqtKind]
             SELECTFROM V[Equipment*EqtKind];Delta

            DELETE FROM maEqtKind[ManagerApproval*EqtKind]
             SELECTFROM V[ManagerApproval*EqtKind];Delta

    (MAINTAINING -(eqtKind~;eqtKind) \/ I[EqtKind] FROM UNI eqtKind::Equipment*EqtKind)
    (MAINTAINING -I[Equipment] \/ eqtKind;eqtKind~ FROM TOT eqtKind::Equipment*EqtKind)
```

<------End Derivation --

```
    ON INSERT Delta IN Isn{detyp=EqtStatus} EXECUTE    -- (ECA rule 71)
    BLOCK
    (CANNOT CHANGE -I[EqtStatus] \/ 'Lost'[EqtStatus] \/ 'Not functional'[EqtStatus]
```

------ Derivation ----->

```
    BLOCK
    (CANNOT CHANGE -I[EqtStatus] \/ 'Lost'[EqtStatus] \/ 'Not functional'[EqtStatus] \/ '
```

<------End Derivation --

```
      ON DELETE Delta FROM Isn{detyp=EqtStatus} EXECUTE    -- (ECA rule 72)
      BLOCK
      (CANNOT CHANGE 'Functional'[EqtStatus] FROM Allowed equipment statuses)
      (CANNOT CHANGE 'Not functional'[EqtStatus] FROM Allowed equipment statuses)
      (CANNOT CHANGE 'Lost'[EqtStatus] FROM Allowed equipment statuses)
```

------ Derivation ----->

```
BLOCK
(CANNOT CHANGE 'Functional'[EqtStatus] FROM Allowed equipment statuses)
(CANNOT CHANGE 'Not functional'[EqtStatus] FROM Allowed equipment statuses)
(CANNOT CHANGE 'Lost'[EqtStatus] FROM Allowed equipment statuses)
```

<------End Derivation --

```
      ON DELETE Delta FROM Isn{detyp=EqtMake} EXECUTE    -- (ECA rule 74)
      ONE OF DELETE FROM eqtMake[Equipment*EqtMake]
              SELECTFROM eqtMake;(-I[EqtMake] /\ eqtMake~;eqtMake)

             (TO MAINTAIN  -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipme
             DELETE FROM eqtMake[Equipment*EqtMake]
              SELECTFROM V[Equipment*EqtMake];Delta

      (MAINTAINING -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*EqtMa
      (MAINTAINING -I[Equipment] \/ eqtMake;eqtMake~ FROM TOT eqtMake::Equipment*EqtMa
```

------ Derivation ----->

```
ONE OF DELETE FROM eqtMake[Equipment*EqtMake]
        SELECTFROM eqtMake;(-I[EqtMake] /\ eqtMake~;eqtMake)

       (TO MAINTAIN  -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*Eq
       DELETE FROM eqtMake[Equipment*EqtMake]
        SELECTFROM V[Equipment*EqtMake];Delta

(MAINTAINING -(eqtMake~;eqtMake) \/ I[EqtMake] FROM UNI eqtMake::Equipment*EqtMake)
(MAINTAINING -I[Equipment] \/ eqtMake;eqtMake~ FROM TOT eqtMake::Equipment*EqtMake)
```

<------End Derivation --

```
ON INSERT Delta IN Isn{detyp=EqtType} EXECUTE    -- (ECA rule 75)
ALL of INSERT INTO typeApprovedProp[EqtType*EqtType]
        SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOf

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
       ONE OF INSERT INTO typeApprovedProp[EqtType*EqtType]
               SELECTFROM I[EqtType] /\ -typeApprovedProp /\ -(typeSecReqt;-type

              (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-type
              ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[EqtType] /\ -typeAppro
                     THEN INSERT INTO typeSecReqt[EqtType*SecRequirement]
                           SELECTFROM 'a'[EqtType]*'b'[SecRequirement]

                          (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSe
                     PICK a,b FROM typeSecReqt~;(I[EqtType] /\ -typeApprovedProp
                     THEN DELETE FROM typeSatReqt[EqtType*SecRequirement]
                           SELECTFROM 'b'[EqtType]*'a'[SecRequirement]

                          (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSe
              (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeS
              NEW x:SecRequirement;
                ALL of INSERT INTO typeSecReqt[EqtType*SecRequirement]
                        SELECTFROM (I[EqtType] /\ -typeApprovedProp /\ -(typeSec

                       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecR
                       DELETE FROM typeSatReqt[EqtType*SecRequirement]
                        SELECTFROM (I[EqtType] /\ -typeApprovedProp~ /\ -(-typeS

                       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecR
                (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typ
              (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeS
       (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM i
```

------ Derivation ----->

```
ALL of INSERT INTO typeApprovedProp[EqtType*EqtType]
        SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
       ONE OF INSERT INTO typeApprovedProp[EqtType*EqtType]
               SELECTFROM I[EqtType] /\ -typeApprovedProp /\ -(typeSecReqt;-typeSatRe

              (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatRe
              ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[EqtType] /\ -typeApprovedP
                     THEN INSERT INTO typeSecReqt[EqtType*SecRequirement]
```

```
                           SELECTFROM 'a'[EqtType]*'b'[SecRequirement]

                          (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReq
                     PICK a,b FROM typeSecReqt~;(I[EqtType] /\ -typeApprovedProp /\ -
                     THEN DELETE FROM typeSatReqt[EqtType*SecRequirement]
                           SELECTFROM 'b'[EqtType]*'a'[SecRequirement]

                          (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReq
              (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReq
              NEW x:SecRequirement;
                ALL of INSERT INTO typeSecReqt[EqtType*SecRequirement]
                        SELECTFROM (I[EqtType] /\ -typeApprovedProp /\ -(typeSecReqt;

                       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-
                       DELETE FROM typeSatReqt[EqtType*SecRequirement]
                        SELECTFROM (I[EqtType] /\ -typeApprovedProp~ /\ -(-typeSatReq

                       (TO MAINTAIN  -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-
                (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatR
              (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReq
       (MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
(MAINTAINING -I[EqtType] \/ typeApprovedProp \/ typeSecReqt;-typeSatReqt~ FROM instyp
```

<------End Derivation --

```
ON DELETE Delta FROM Isn{detyp=EqtType} EXECUTE    -- (ECA rule 76)
ONE OF DELETE FROM typeApprovedProp[EqtType*EqtType]
        SELECTFROM (-I[EqtType] /\ typeApprovedProp;typeApprovedBySecOff;'Yes'[Y

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answe
       DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
        SELECTFROM typeApprovedProp~;(-I[EqtType] /\ typeApprovedProp;typeApprov

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answe
       DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
        SELECTFROM (-I[EqtType] /\ typeApprovedBySecOff;'Yes'[Yes/No answer];typ

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answe
       DELETE FROM typeApprovedProp[EqtType*EqtType]
        SELECTFROM -I[EqtType] /\ typeApprovedProp;typeApprovedBySecOff;'Yes'[Ye

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answe
       DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
        SELECTFROM (-I[EqtType] /\ typeApprovedBySecOff;'Yes'[Yes/No answer];typ

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedByS
       DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
```

```
         SELECTFROM typeApprovedProp;(-I[EqtType] /\ typeApprovedProp~;typeApprov

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOf

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM -I[EqtType] /\ typeApprovedBySecOff;'Yes'[Yes/No answer];type

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM -I[EqtType] /\ typeApprovedProp

        (TO MAINTAIN  -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
        DELETE FROM eqtType[Equipment*EqtType]
         SELECTFROM eqtType;(-I[EqtType] /\ eqtType~;eqtType)

        (TO MAINTAIN  -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipme
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM (-I[EqtType] /\ typeApprovedProp;typeApprovedProp);typeApprov

        (TO MAINTAIN  -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM typeApprovedProp~;(-I[EqtType] /\ typeApprovedProp;typeApprov

        (TO MAINTAIN  -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI
        DELETE FROM eqtType[Equipment*EqtType]
         SELECTFROM V[Equipment*EqtType];Delta

        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM Delta;V[EqtType*EqtType]

        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM V[EqtType*EqtType];Delta

        DELETE FROM typeSecReqt[EqtType*SecRequirement]
         SELECTFROM Delta;V[EqtType*SecRequirement]

        DELETE FROM typeSatReqt[EqtType*SecRequirement]
         SELECTFROM Delta;V[EqtType*SecRequirement]

        DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
         SELECTFROM Delta;V[EqtType*Yes/No answer]

(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
(MAINTAINING -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
(MAINTAINING -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*EqtTy
(MAINTAINING -I[Equipment] \/ eqtType;eqtType~ FROM TOT eqtType::Equipment*EqtTy
(MAINTAINING -typeApprovedProp \/ I[EqtType] FROM ASY typeApprovedProp::EqtType*
(MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI typeAppr
(MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM INJ typeAppr
```

------ Derivation ----->

```
 ONE OF DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM (-I[EqtType] /\ typeApprovedProp;typeApprovedBySecOff;'Yes'[Yes/No

        (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
        DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
         SELECTFROM typeApprovedProp~;(-I[EqtType] /\ typeApprovedProp;typeApprovedByS

        (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
        DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
         SELECTFROM (-I[EqtType] /\ typeApprovedBySecOff;'Yes'[Yes/No answer];typeAppr

        (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM -I[EqtType] /\ typeApprovedProp;typeApprovedBySecOff;'Yes'[Yes/No

        (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
        DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
         SELECTFROM (-I[EqtType] /\ typeApprovedBySecOff;'Yes'[Yes/No answer];typeAppr

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
        DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
         SELECTFROM typeApprovedProp;(-I[EqtType] /\ typeApprovedProp~;typeApprovedByS

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~;(-

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM -I[EqtType] /\ typeApprovedBySecOff;'Yes'[Yes/No answer];typeAppro

        (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM -I[EqtType] /\ typeApprovedProp

        (TO MAINTAIN  -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
        DELETE FROM eqtType[Equipment*EqtType]
         SELECTFROM eqtType;(-I[EqtType] /\ eqtType~;eqtType)

        (TO MAINTAIN  -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*Eq
        DELETE FROM typeApprovedProp[EqtType*EqtType]
         SELECTFROM (-I[EqtType] /\ typeApprovedProp;typeApprovedProp);typeApprovedPro
```

```
            (TO MAINTAIN  -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI type
            DELETE FROM typeApprovedProp[EqtType*EqtType]
             SELECTFROM typeApprovedProp~;(-I[EqtType] /\ typeApprovedProp;typeApprovedPro

            (TO MAINTAIN  -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI type
            DELETE FROM eqtType[Equipment*EqtType]
             SELECTFROM V[Equipment*EqtType];Delta

            DELETE FROM typeApprovedProp[EqtType*EqtType]
             SELECTFROM Delta;V[EqtType*EqtType]

            DELETE FROM typeApprovedProp[EqtType*EqtType]
             SELECTFROM V[EqtType*EqtType];Delta

            DELETE FROM typeSecReqt[EqtType*SecRequirement]
             SELECTFROM Delta;V[EqtType*SecRequirement]

            DELETE FROM typeSatReqt[EqtType*SecRequirement]
             SELECTFROM Delta;V[EqtType*SecRequirement]

            DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
             SELECTFROM Delta;V[EqtType*Yes/No answer]

     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
     (MAINTAINING -typeApprovedProp \/ I[EqtType] FROM deltypeApprovedProp)
     (MAINTAINING -(eqtType~;eqtType) \/ I[EqtType] FROM UNI eqtType::Equipment*EqtType)
     (MAINTAINING -I[Equipment] \/ eqtType;eqtType~ FROM TOT eqtType::Equipment*EqtType)
     (MAINTAINING -typeApprovedProp \/ I[EqtType] FROM ASY typeApprovedProp::EqtType*EqtTy
     (MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM UNI typeApproved
     (MAINTAINING -(typeApprovedProp;typeApprovedProp) \/ I[EqtType] FROM INJ typeApproved
```

<------End Derivation --

```
          ON DELETE Delta FROM Isn{detyp=EqtSerial} EXECUTE    -- (ECA rule 78)
          ONE OF DELETE FROM eqtSerial[Equipment*EqtSerial]
                  SELECTFROM eqtSerial;(-I[EqtSerial] /\ eqtSerial~;eqtSerial)

                 (TO MAINTAIN  -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial:
                 DELETE FROM eqtSerial[Equipment*EqtSerial]
                  SELECTFROM V[Equipment*EqtSerial];Delta

          (MAINTAINING -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equipme
          (MAINTAINING -I[Equipment] \/ eqtSerial;eqtSerial~ FROM TOT eqtSerial::Equipment
```

------ Derivation ----->

```
ONE OF DELETE FROM eqtSerial[Equipment*EqtSerial]
        SELECTFROM eqtSerial;(-I[EqtSerial] /\ eqtSerial~;eqtSerial)

       (TO MAINTAIN  -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equi
       DELETE FROM eqtSerial[Equipment*EqtSerial]
        SELECTFROM V[Equipment*EqtSerial];Delta

(MAINTAINING -(eqtSerial~;eqtSerial) \/ I[EqtSerial] FROM UNI eqtSerial::Equipment*Eq
(MAINTAINING -I[Equipment] \/ eqtSerial;eqtSerial~ FROM TOT eqtSerial::Equipment*EqtS
```

<------End Derivation --

```
ON INSERT Delta IN Isn{detyp=ManagerApproval} EXECUTE    -- (ECA rule 79)
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[ManagerApproval] /\ -(maEmpl
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;ma
              PICK a,b FROM maEmployee~;(I[ManagerApproval] /\ -(maEmployee;empl
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[Employee
                                 THEN INSERT INTO emplManager[Employee*Employee]
                                       SELECTFROM 'a'[Employee]*'b'[Employee]

                                      (TO MAINTAIN  -I[ManagerApproval] \/ maEmpl
                                 PICK a,b FROM emplManager~;('a'[Employee]*'b'[Ma
                                 THEN INSERT INTO maManager[ManagerApproval*Empl
                                       SELECTFROM 'b'[ManagerApproval]*'a'[Emplo

                                      (TO MAINTAIN  -I[ManagerApproval] \/ maEmpl
                          (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplMana
                          NEW x:Employee;
                            ALL of INSERT INTO emplManager[Employee*Employee]
                                    SELECTFROM 'a'[Employee]*'b'[ManagerApproval

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmploye
                                   INSERT INTO maManager[ManagerApproval*Employee
                                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmploye
                            (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplMa
                          (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplMana
                   (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maM
       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM
       NEW x:Employee;
         ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                 SELECTFROM (I[ManagerApproval] /\ -(maEmployee;emplManager;maMa
                (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maMa
                ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Employee]*
                              THEN INSERT INTO emplManager[Employee*Employee]
                                    SELECTFROM 'a'[Employee]*'b'[Employee]

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmploy
                              PICK a,b FROM emplManager~;('x'[Employee]*(I[Manag
                              THEN INSERT INTO maManager[ManagerApproval*Employe
                                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmploy
                       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManage
                       NEW x:Employee;
                         ALL of INSERT INTO emplManager[Employee*Employee]
                                 SELECTFROM 'x'[Employee]*(I[ManagerApproval] /\

                                (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;
                                INSERT INTO maManager[ManagerApproval*Employee]
                                 SELECTFROM (I[ManagerApproval] /\ -(maManager;e

                                (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;
                         (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplMana
                       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManage
                (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maMan
         (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ F
       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FRO
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[ManagerApproval] /\ -(maEmpl
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;I[Employee];m
              PICK a,b FROM maEmployee~;(I[ManagerApproval] /\ -(maEmployee;maEm
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;I[Employee];m
       (MAINTAINING -I[ManagerApproval] \/ maEmployee;I[Employee];maEmployee~ FR
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[ManagerApproval] /\ -(maMana
              THEN INSERT INTO maManager[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maManager;I[Employee];ma
              PICK a,b FROM maManager~;(I[ManagerApproval] /\ -(maManager;maMana
              THEN INSERT INTO maManager[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maManager;I[Employee];ma
       (MAINTAINING -I[ManagerApproval] \/ maManager;I[Employee];maManager~ FROM
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integ
(MAINTAINING -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::Manag
(MAINTAINING -I[ManagerApproval] \/ maEmployee;maEmployee~ FROM TOT maEmployee::
(MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerA
(MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::Man
```

------ Derivation ----->

```
ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[ManagerApproval] /\ -(maEmployee;
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maMana
              PICK a,b FROM maEmployee~;(I[ManagerApproval] /\ -(maEmployee;emplManag
              THEN ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('a'[Employee]*'b
                                 THEN INSERT INTO emplManager[Employee*Employee]
                                       SELECTFROM 'a'[Employee]*'b'[Employee]

                                      (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee
                                 PICK a,b FROM emplManager~;('a'[Employee]*'b'[Manage
                                 THEN INSERT INTO maManager[ManagerApproval*Employee]
                                       SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                                      (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee
                          (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;
                          NEW x:Employee;
                            ALL of INSERT INTO emplManager[Employee*Employee]
                                    SELECTFROM 'a'[Employee]*'b'[ManagerApproval]*'x'

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;em
                                   INSERT INTO maManager[ManagerApproval*Employee]
                                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]*'x'

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;em
                            (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManage
                          (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;
                   (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManag
       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM Man
       NEW x:Employee;
         ALL of INSERT INTO maEmployee[ManagerApproval*Employee]
                 SELECTFROM (I[ManagerApproval] /\ -(maEmployee;emplManager;maManager

                (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplManager;maManager
                ONE OF ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM ('x'[Employee]*(I[Ma
                              THEN INSERT INTO emplManager[Employee*Employee]
                                    SELECTFROM 'a'[Employee]*'b'[Employee]

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;em
                              PICK a,b FROM emplManager~;('x'[Employee]*(I[ManagerApp
                              THEN INSERT INTO maManager[ManagerApproval*Employee]
                                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;em
                       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maM
                       NEW x:Employee;
                         ALL of INSERT INTO emplManager[Employee*Employee]
                                 SELECTFROM 'x'[Employee]*(I[ManagerApproval] /\ -(ma

                                (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplM
                                INSERT INTO maManager[ManagerApproval*Employee]
                                 SELECTFROM (I[ManagerApproval] /\ -(maManager;emplMa

                                (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;emplM
                         (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;m
                       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maM
                (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~
         (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM M
       (MAINTAINING -I[ManagerApproval] \/ maEmployee;emplManager;maManager~ FROM Man
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[ManagerApproval] /\ -(maEmployee;
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;I[Employee];maEmpl
              PICK a,b FROM maEmployee~;(I[ManagerApproval] /\ -(maEmployee;maEmploye
              THEN INSERT INTO maEmployee[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maEmployee;I[Employee];maEmpl
       (MAINTAINING -I[ManagerApproval] \/ maEmployee;I[Employee];maEmployee~ FROM UN
       ONE NONEMPTY ALTERNATIVE OF PICK a,b FROM (I[ManagerApproval] /\ -(maManager;m
              THEN INSERT INTO maManager[ManagerApproval*Employee]
                    SELECTFROM 'a'[ManagerApproval]*'b'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maManager;I[Employee];maManag
              PICK a,b FROM maManager~;(I[ManagerApproval] /\ -(maManager;maManager~)
              THEN INSERT INTO maManager[ManagerApproval*Employee]
                    SELECTFROM 'b'[ManagerApproval]*'a'[Employee]

                   (TO MAINTAIN  -I[ManagerApproval] \/ maManager;I[Employee];maManag
       (MAINTAINING -I[ManagerApproval] \/ maManager;I[Employee];maManager~ FROM UNI
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maManager) \/ emplManager FROM Manager approval integrity)
(MAINTAINING -(maEmployee~;maEmployee) \/ I[Employee] FROM UNI maEmployee::ManagerApp
(MAINTAINING -I[ManagerApproval] \/ maEmployee;maEmployee~ FROM TOT maEmployee::Manag
(MAINTAINING -(maManager~;maManager) \/ I[Employee] FROM UNI maManager::ManagerApprov
(MAINTAINING -I[ManagerApproval] \/ maManager;maManager~ FROM TOT maManager::ManagerA
```

<------End Derivation --

```
         ON DELETE Delta FROM Isn{detyp=ManagerApproval} EXECUTE    -- (ECA rule 80)
         ALL of DELETE FROM maEmployee[ManagerApproval*Employee]
                 SELECTFROM Delta;V[ManagerApproval*Employee]

                DELETE FROM maManager[ManagerApproval*Employee]
                 SELECTFROM Delta;V[ManagerApproval*Employee]

                DELETE FROM maEqtKind[ManagerApproval*EqtKind]
                 SELECTFROM Delta;V[ManagerApproval*EqtKind]
```

------ Derivation ----->

```
    ALL of DELETE FROM maEmployee[ManagerApproval*Employee]
            SELECTFROM Delta;V[ManagerApproval*Employee]

           DELETE FROM maManager[ManagerApproval*Employee]
            SELECTFROM Delta;V[ManagerApproval*Employee]

           DELETE FROM maEqtKind[ManagerApproval*EqtKind]
            SELECTFROM Delta;V[ManagerApproval*EqtKind]
```

<------End Derivation --

```
         ON DELETE Delta FROM Isn{detyp=SecRequirement} EXECUTE    -- (ECA rule 82)
         ALL of DELETE FROM eqtSecReqt[Equipment*SecRequirement]
                 SELECTFROM V[Equipment*SecRequirement];Delta

                DELETE FROM eqtSatReqt[Equipment*SecRequirement]
                 SELECTFROM V[Equipment*SecRequirement];Delta

                DELETE FROM typeSecReqt[EqtType*SecRequirement]
                 SELECTFROM V[EqtType*SecRequirement];Delta

                DELETE FROM typeSatReqt[EqtType*SecRequirement]
                 SELECTFROM V[EqtType*SecRequirement];Delta
```

------ Derivation ----->

```
    ALL of DELETE FROM eqtSecReqt[Equipment*SecRequirement]
            SELECTFROM V[Equipment*SecRequirement];Delta
```

```
       DELETE FROM eqtSatReqt[Equipment*SecRequirement]
        SELECTFROM V[Equipment*SecRequirement];Delta

       DELETE FROM typeSecReqt[EqtType*SecRequirement]
        SELECTFROM V[EqtType*SecRequirement];Delta

       DELETE FROM typeSatReqt[EqtType*SecRequirement]
        SELECTFROM V[EqtType*SecRequirement];Delta
```

<------End Derivation --

```
     ON INSERT Delta IN Isn{detyp=Yes/No answer} EXECUTE    -- (ECA rule 83)
     ALL of INSERT INTO eqtApprovedProp[Equipment*Equipment]
             SELECTFROM eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~

            (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
            INSERT INTO Isn{detyp=Equipment}
             SELECTFROM (eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqt

            (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer]
            (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySec
            INSERT INTO typeApprovedProp[EqtType*EqtType]
             SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOf

            (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
            INSERT INTO Isn{detyp=EqtType}
             SELECTFROM (typeApprovedProp;typeApprovedBySecOff;'Yes'[Yes/No answer];t

            (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answe
            (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySe
     (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\
     (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\
     (MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\
     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
     (MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /
```

------ Derivation ----->

```
ALL of INSERT INTO eqtApprovedProp[Equipment*Equipment]
        SELECTFROM eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I

       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~
       INSERT INTO Isn{detyp=Equipment}
        SELECTFROM (eqtApprovedProp;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtAppro
       (TO MAINTAIN  -(eqtApprovedProp~;eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtA
       (TO MAINTAIN  -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~;
       INSERT INTO typeApprovedProp[EqtType*EqtType]
        SELECTFROM typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\

       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
       INSERT INTO Isn{detyp=EqtType}
        SELECTFROM (typeApprovedProp;typeApprovedBySecOff;'Yes'[Yes/No answer];typeAp

       (TO MAINTAIN  -(typeApprovedProp~;typeApprovedBySecOff;'Yes'[Yes/No answer];ty
       (TO MAINTAIN  -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff
(MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
(MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
(MAINTAINING -(eqtApprovedBySecOff;'Yes'[Yes/No answer];eqtApprovedBySecOff~ /\ I[Equ
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
(MAINTAINING -(typeApprovedBySecOff;'Yes'[Yes/No answer];typeApprovedBySecOff~ /\ I[E
```

<------End Derivation --

```
     ON DELETE Delta FROM Isn{detyp=Yes/No answer} EXECUTE    -- (ECA rule 84)
     ALL of DELETE FROM eqtApprovedProp[Equipment*Equipment]
             SELECTFROM -(eqtSecReqt~ \ eqtSatReqt~) /\ -(eqtApprovedBySecOff;'Yes'[Y

            (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprove
            DELETE FROM typeApprovedProp[EqtType*EqtType]
             SELECTFROM -(typeSecReqt~ \ typeSatReqt~) /\ -(typeApprovedBySecOff;'Yes

            (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeAppr
            DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
             SELECTFROM V[Equipment*Yes/No answer];Delta

            DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
             SELECTFROM V[EqtType*Yes/No answer];Delta

     (MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOf
     (MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedByS
```

------ Derivation ----->

```
ALL of DELETE FROM eqtApprovedProp[Equipment*Equipment]
        SELECTFROM -(eqtSecReqt~ \ eqtSatReqt~) /\ -(eqtApprovedBySecOff;'Yes'[Yes/No

       (TO MAINTAIN  -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySe
       DELETE FROM typeApprovedProp[EqtType*EqtType]
        SELECTFROM -(typeSecReqt~ \ typeSatReqt~) /\ -(typeApprovedBySecOff;'Yes'[Yes
       (TO MAINTAIN  -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApproved
       DELETE FROM eqtApprovedBySecOff[Equipment*Yes/No answer]
        SELECTFROM V[Equipment*Yes/No answer];Delta

       DELETE FROM typeApprovedBySecOff[EqtType*Yes/No answer]
        SELECTFROM V[EqtType*Yes/No answer];Delta

(MAINTAINING -eqtApprovedProp \/ eqtSecReqt~ \ eqtSatReqt~ \/ eqtApprovedBySecOff;'Ye
(MAINTAINING -typeApprovedProp \/ typeSecReqt~ \ typeSatReqt~ \/ typeApprovedBySecOff
```

<------End Derivation --

```
     ON INSERT Delta IN Isn{detyp=Status} EXECUTE    -- (ECA rule 85)
     ALL of INSERT INTO emplStatus[Employee*Status]
             SELECTFROM (noNecessaryEqtHasBeenIssued~;emplStatus;'Grey'[Status] /\ al

            (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued /\
            (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /\ 
            INSERT INTO Isn{detyp=Status}
             SELECTFROM ('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;emplS

            (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;em
            (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Grey'
     (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
     (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
     (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
     (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
```

------ Derivation ----->

```
ALL of INSERT INTO emplStatus[Employee*Status]
        SELECTFROM (noNecessaryEqtHasBeenIssued~;emplStatus;'Grey'[Status] /\ allNece

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued /\ 'Gre
       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status] /\ allNe
       INSERT INTO Isn{detyp=Status}
        SELECTFROM ('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;emplSta
       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Stat
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
```

<------End Derivation --

```
ON DELETE Delta FROM Isn{detyp=Status} EXECUTE    -- (ECA rule 86)
ALL of DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Yellow'[Status];emplStatus~) /\ -noNecessaryEqt

       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;emplStatus) \/ V[Employ

       (TO MAINTAIN  -(emplStatus~;emplStatus) \/ I[Status] FROM UNI emplStatus:
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Black'[Status];emplStatus~) /\ -needsToR

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Black'[Status];emplStatus~) /\ -needsToR

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -(emplStatus;'Black'[Status];emplStatus~) /\ -needsToR

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssue
       ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Green'[Status];emplStatus~) /\ -noNecess

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -(emplStatus;'Green'[Status];emplStatus~) /\ -noNecess

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ em
       (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Red'[Status];emplStatus~) /\ -allNecessa

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ empl
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -(emplStatus;'Red'[Status];emplStatus~) /\ -allNecessa

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ empl
       (MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessar

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessar

              (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
              DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessary

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessary

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasB
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssue
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;(-(emplStatus;'Grey'[Statu

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];emplStatus~) /\ emplStatu

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;(-(emplStatus;'Grey'[Stat

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];emplStatus~) /\ emplStatu

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;(-(emplStatus;'Grey'[Status]) /\ noNe

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];emplStatus~) /\ emplStatu

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM -(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenI

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplSta
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;all
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;emplStatus;(-I[Status] /\ 

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;emplStatus;'Grey'[Status]

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
```

```
        SELECTFROM allNecessaryEqtHasBeenIssued;emplStatus;(-I[Status] /\

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;emplStatus;'Grey'[Status

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;emplStatus;(-I[Status] /\ emplStatus~

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;emplStatus;'Grey'[Status];(-I[Status

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIs
(MAINTAINING -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;empl
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;emplStatus;(-'Grey'[Status

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessary

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;emplStatus;(-'Grey'[Statu

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;emplStatus;(-'Grey'[Statu

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessary
       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;emplStatus;(-'Grey'[Stat

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;emplStatus;(-'Grey'[Status] /\ emplSt

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryE

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;emplStatus;(-'Grey'[Status] /\ emplSt

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryE

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryE

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ emplS
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status];

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;emplStatus;(-I[Status] /\

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status]

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;emplStatus;(-I[Status] /
       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;emplStatus;'Grey'[Status];(-I[Status]

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;emplStatus;(-I[Status] /\ emplStatus

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM (-(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeen

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;(-(emplStatus;'Grey'[Stat

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM (-(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeen

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;(-(emplStatus;'Grey'[Sta

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM (-(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeen

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;(-(emplStatus;'Grey'[Status]) /\ noN

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM -(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenI

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNeces
       (MAINTAINING -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryEqtH
       ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
               SELECTFROM -(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessa

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
              DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM -(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessa

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessa

              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /
      (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Empl
       ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM -(emplStatus;'Orange'[Status];emplStatus~) /\ -noNeces

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Or
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -(emplStatus;'Orange'[Status];emplStatus~) /\ -noNeces

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Or
       (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[St
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ I[
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'[S
(MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecess
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ ne
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee])
(MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];e
(MAINTAINING -(emplStatus~;emplStatus) \/ I[Status] FROM UNI emplStatus::Employe
```

------ Derivation ----->

```
ALL of DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Yellow'[Status];emplStatus~) /\ -noNecessaryEqtHasBe

       (TO MAINTAIN  -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNec
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;emplStatus) \/ V[Employee*St

       (TO MAINTAIN  -(emplStatus~;emplStatus) \/ I[Status] FROM UNI emplStatus::Empl
       ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
```

```
        SELECTFROM -(emplStatus;'Black'[Status];emplStatus~) /\ -needsToReturn

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(emplStatus;'Black'[Status];emplStatus~) /\ -needsToReturn

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Black'[Status];emplStatus~) /\ -needsToReturn

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\
ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(emplStatus;'Green'[Status];emplStatus~) /\ -noNecessaryEq

       (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Green'[Status];emplStatus~) /\ -noNecessaryEq

       (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplSta
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Gre
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(emplStatus;'Red'[Status];emplStatus~) /\ -allNecessaryEqt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStat
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Red'[Status];emplStatus~) /\ -allNecessaryEqt

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStat
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessaryEqtH

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessaryEqtH

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessaryEqtH

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Grey'[Status];emplStatus~) /\ noNecessaryEqtH

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIs
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;(-(emplStatus;'Grey'[Status]) /
       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];emplStatus~) /\ emplStatus~;no

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;(-(emplStatus;'Grey'[Status])

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];emplStatus~) /\ emplStatus~;no

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;(-(emplStatus;'Grey'[Status]) /\ noNecessa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;(-('Grey'[Status];emplStatus~) /\ emplStatus~;no

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM -(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenIssued

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued /\ emplStatus~;allNeces
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;emplStatus;(-I[Status] /\ emplS

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status];emp

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;emplStatus;'Grey'[Status];(-I[

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;emplStatus;(-I[Status] /\ empl

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status];emp

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;emplStatus;'Grey'[Status];(-I

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;emplStatus;(-I[Status] /\ emplStatus~;noNe

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status];emp

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;emplStatus;'Grey'[Status];(-I[Status] /\

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHasBeen

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status];emp

       (TO MAINTAIN  -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;
(MAINTAINING -('Grey'[Status];emplStatus~;noNecessaryEqtHasBeenIssued;emplStat
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;emplStatus;(-'Grey'[Status] /\

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;emplStatus;(-'Grey'[Status] /\

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;emplStatus;(-'Grey'[Status] /\

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;emplStatus;(-'Grey'[Status] /

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;emplStatus;(-'Grey'[Status] /\ emplStatus~

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM needsToReturnEqt[Employee*Employee]
```

```
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;emplStatus;(-'Grey'[Status] /\ emplStatus

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-'Grey'[Status] /\ emplStatus~;noNecessaryEqtHa

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ e
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus /\ emplStatu
ONE OF DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status];(-I[S

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHasBeen

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;emplStatus;(-I[Status] /\ empl

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Status];(-I[

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHasBeen

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;emplStatus;(-I[Status] /\ emp

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt;emplStatus;'Grey'[Status];(-I[Status] /\ '

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHasBeen

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;emplStatus;(-I[Status] /\ emplStatus~;noN
       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;'Grey'[Status];(-I[Status] /\ 'Grey'[Status];emp

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM emplStatus;(-I[Status] /\ emplStatus~;noNecessaryEqtHasBeen

       (TO MAINTAIN  -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Gre
(MAINTAINING -(emplStatus~;noNecessaryEqtHasBeenIssued;emplStatus;'Grey'[Statu
ONE OF DELETE FROM noNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM (-(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenIssue

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM noNecessaryEqtHasBeenIssued~;(-(emplStatus;'Grey'[Status])

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM (-(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenIssue

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM allNecessaryEqtHasBeenIssued~;(-(emplStatus;'Grey'[Status])

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM (-(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenIssue

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM needsToReturnEqt~;(-(emplStatus;'Grey'[Status]) /\ noNecess

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
       DELETE FROM emplStatus[Employee*Status]
        SELECTFROM -(emplStatus;'Grey'[Status]) /\ noNecessaryEqtHasBeenIssued

       (TO MAINTAIN  -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryE
(MAINTAINING -(noNecessaryEqtHasBeenIssued;emplStatus /\ allNecessaryEqtHasBee
ONE OF DELETE FROM allNecessaryEqtHasBeenIssued[Employee*Employee]
        SELECTFROM -(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessaryEqt

       (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
       DELETE FROM needsToReturnEqt[Employee*Employee]
        SELECTFROM -(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessaryEqt

       (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
       DELETE FROM Isn{detyp=Employee}
        SELECTFROM -(emplStatus;'Blue'[Status];emplStatus~) /\ -noNecessaryEqt
```

```
              (TO MAINTAIN  -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[E
       (MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee]
       ONE OF DELETE FROM needsToReturnEqt[Employee*Employee]
               SELECTFROM -(emplStatus;'Orange'[Status];emplStatus~) /\ -noNecessaryE

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'
              DELETE FROM Isn{detyp=Employee}
               SELECTFROM -(emplStatus;'Orange'[Status];emplStatus~) /\ -noNecessaryE

              (TO MAINTAIN  -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'
       (MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status]
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ I[Emplo
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Green'[Sta
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ I[Employee]) \/ emplStatus;'Red'[Status
(MAINTAINING -I[Employee] \/ emplStatus;'Yellow'[Status];emplStatus~ \/ noNecessaryEq
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(noNecessaryEqtHasBeenIssued /\ allNecessaryEqtHasBeenIssued /\ needsTo
(MAINTAINING -(allNecessaryEqtHasBeenIssued /\ needsToReturnEqt /\ I[Employee]) \/ em
(MAINTAINING -(needsToReturnEqt /\ I[Employee]) \/ emplStatus;'Orange'[Status];emplSt
(MAINTAINING -(emplStatus~;emplStatus) \/ I[Status] FROM UNI emplStatus::Employee*Sta
```

<------End Derivation --

```
ON DELETE Delta FROM Isn{detyp=SESSION} EXECUTE    -- (ECA rule 88)
ALL of DELETE FROM sessionEmployee[SESSION*Employee]
        SELECTFROM Delta;V[SESSION*Employee]

       DELETE FROM sessionOrgRole[SESSION*OrganizationalRole]
        SELECTFROM Delta;V[SESSION*OrganizationalRole]
```

------ Derivation ----->

```
ALL of DELETE FROM sessionEmployee[SESSION*Employee]
        SELECTFROM Delta;V[SESSION*Employee]

       DELETE FROM sessionOrgRole[SESSION*OrganizationalRole]
        SELECTFROM Delta;V[SESSION*OrganizationalRole]
```

<------End Derivation --

Glossary

Employee a person that has been issued a personal ID-card of Company Inc.. 3

EmployeeName a human readable text that uniquely identifies an employee. 4

EqtKind A class of equipment. 4

Equipment an (identifiable) object that can be moved/taken away with relative ease, and that employees may need to do their job. 4

ManagerApproval an approval, by a manager, for an employee, allowing the employee to use a specific kind of company equipment. 7

OrganizationalRole a set of (related) responsibilities as defined by Company Inc., assigned to employees. 4

SecRequirement the specification of a requirement for some equipment types. 8