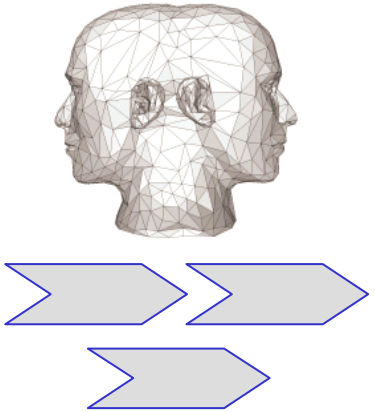
NIFIS

GenericIAM

generic processes for the
Identity- & Access Management



Conclave workshop

2007-06-27 – 28

Arslan Brömme • Andreas Netzer • Horst Walther

Benediktinerabtei zum Hl. Kreuz
Schyrenplatz • 1 85298 Scheyern

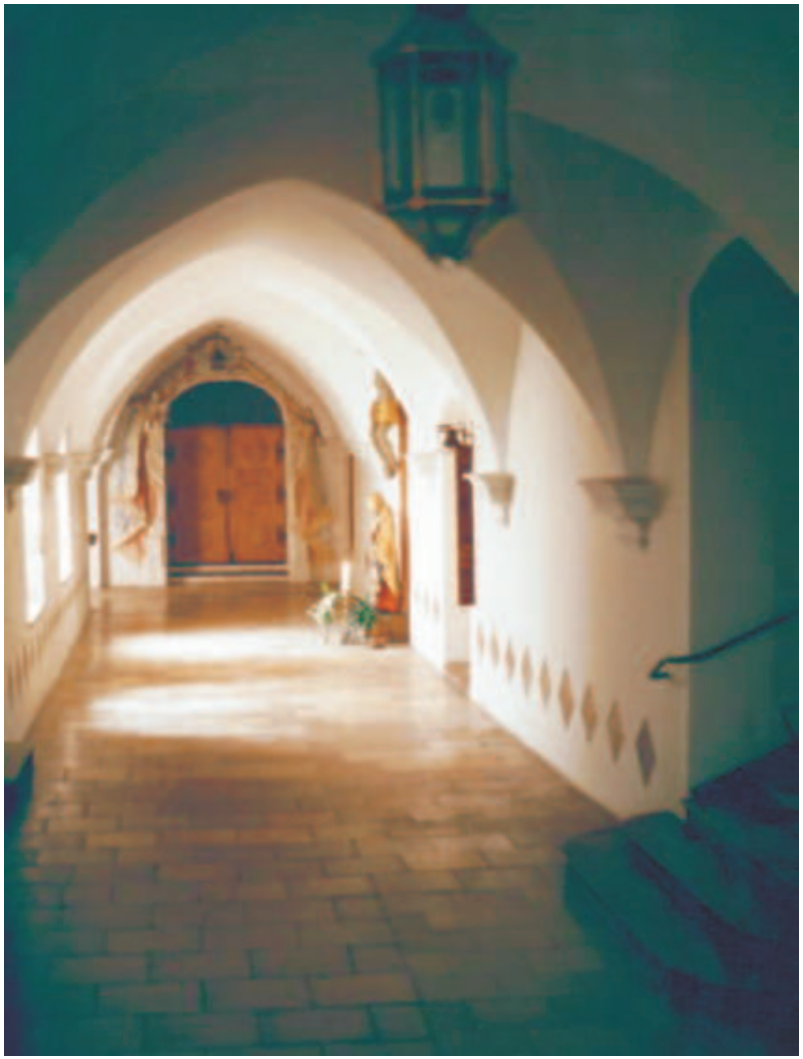
Version 1.0

- Start: Wednesday, 2007-06-27, 09:00
- End: Thursday, 2007-06-28, 17:00

- Despite the fact, that the location offers some leisure time facilities, the major focus will be intensive modelling work. ;-)
- Each participant should be prepared to contribute appropriately. Personal tasks will be assigned in bilateral talks with registered attendees.

The location
Kloster Scheyern

- Location
  - Benediktinerabtei zum Hl. Kreuz
  - Schyrenplatz 1
  - 85298 Scheyern
- Telefon:
  - 08441/ 752 - 0 (Klosterpforte)
  - 08441/ 752 - 230 (Klosterverwaltung)
  - 08441/ 752 - 181 (Kath. Pfarramt)
- Async contacts
  - Telefax: 08441/ 752 - 210
  - e-mail: info@kloster-scheyern.de
  - http://www.kloster-scheyern.de/
- Room prices are ~ 30 € per night.
- Additional costs (e.g. conference room) where sponsored by iC Compas.



Terminology

- The key words *must, must not, required, shall, shall not, should, should not, recommended, may,* and *optional* in this document are to be interpreted as described in IETF RFC 2119 [RFC2119].
- **Attribute**
  - An Attribute is an element in a Request having among its components an attribute name identifier, a data type identifier, and an attribute value.
  - Each Attribute is associated either with one of the subjects (Subject Attribute), the protected resource (Resource Attribute), the action to be taken on the resource (Action Attribute), or the environment of the Request (Environment Attribute).
  - Attributes my be referenced in a policy.
- **junior role**
  - In a role hierarchy, Role A is *junior* to Role B if Role B inherits all the permissions associated with Role A.
- **multi-role permissions**
  - A set of permissions for which a user must hold more than one role simultaneously in order to gain access.
- **PDP**
  - Policy Decision Point. An entity that evaluates an access request against one or more policies to produce an access decision.
- **Permission**
  - The ability or right to perform some action on some resource, possibly only under certain specified conditions.
- **RBAC**
  - Role based access control. A model for controlling access to resources where permitted actions on resources are identified with roles rather than with individual subject identities.
- **Role**
  - A job function within the context of an organization that has associated semantics regarding the authority and responsibility conferred on the user assigned to the role [RBAC].
- **senior role**
  - In a role hierarchy, Role A is *senior* to Role B if Role A inherits all the permissions associated with Role B.
- **Policy**
  - A set of rules indicating which subjects are permitted to access which resources using which actions under which conditions.

Policies

4 types of policies

1. **Role**
   - A “policy set” that associates holders of a given role attribute with the actual permissions associated with the given role.
   - A “target” specification of a role limits the applicability to subjects holding the given role attribute.
   - Each Role references a single corresponding set of permissions but does not contain any other policies.

2. **Permission**:
   - A “policy set” that contains the actual permissions associated with a given role.
   - It contains policy elements and rules that describe the resources and actions that subjects are permitted to access, along with any further conditions on that access, such as time of day.
   - A given permission may also contain references to permissions associated with other roles that are *junior* to the given role, allowing the given permission to inherit all permissions associated with the role of the referenced Permission .
   - The target element of a Permission must not limit the subjects to which the is applicable.

3. **Separation of Duty**
   - A “policy set” that defines restrictions on the set of roles that can be exercised by a given Subject.
   - Such a “policy set” contains policies and rules that specify the role set restrictions.
   - The Separation of Duty also contains references to all the Role instances that are subject to Separation of Duty restrictions.
   - Use of a Separation of Duty is optional.

4. **Role Assignment**
   - a policy or “policy set” that defines which roles can be enabled or assigned to which subjects.
   - It may also specify restrictions on combinations of roles or total number of roles assigned to or enabled for a given subject.
   - This type of policy is used by the entity that assigns role attributes to users or by the entity that enables role attributes during a user's session.
   - Use of a Role Assignment policy or policy set is optional.

Core RBAC

Core RBAC includes the following five basic data elements

1. Users

- **Users** are implemented using Subjects.
- Any of the Subject values may be used, as appropriate.

2. Roles

- **Roles** are expressed using one or more Subject Attributes.
- The set of roles is application and policy domain-specific, and it is very important that different uses of roles not be confused.
- XACML does not define any standard set of roles.
- It is recommended that each application or policy domain agree on and publish a unique set of Attribute-Id values, Data Type values, and Attribute values that will be used for the various roles relevant to that domain.

3. Objects

- **Objects** are expressed using Resources.

4. Operations

- **Operations** are expressed using Actions.

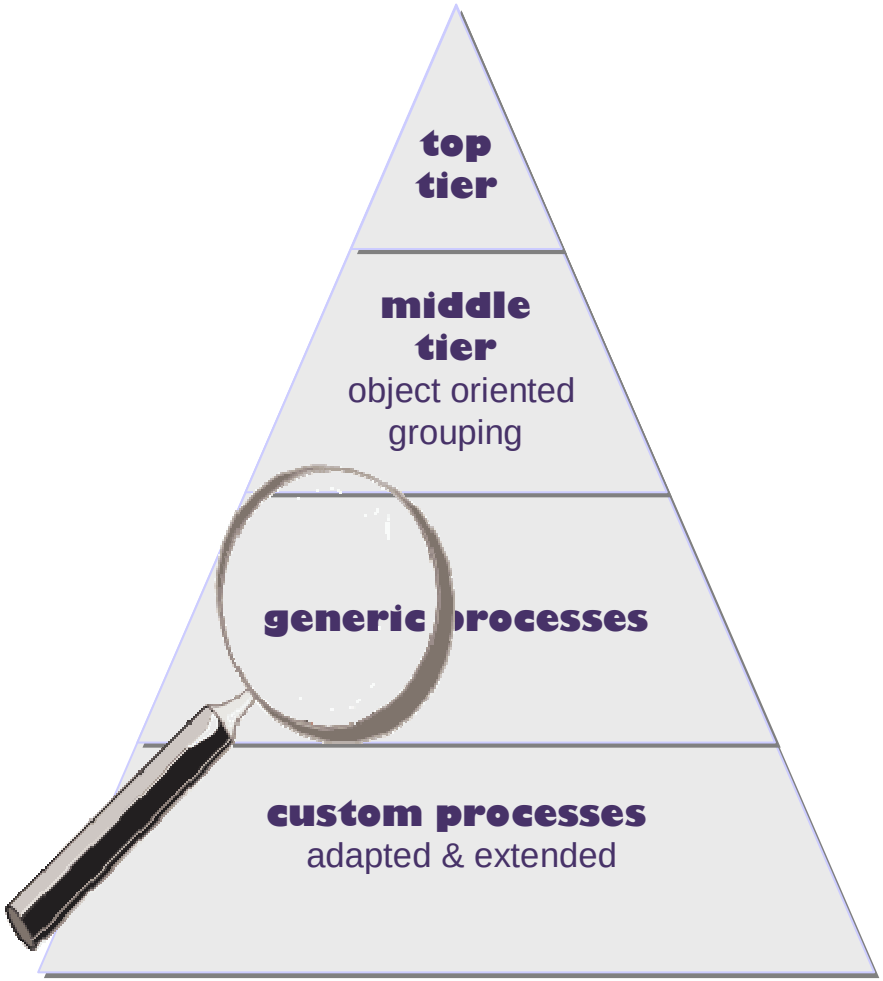
5. Permissions

- **Permissions** are expressed using Role <PolicySet> and Permission <PolicySet> instances as described in previous sections.

- Core RBAC requires support for multiple users per role, multiple roles per user, multiple permissions per role, and multiple roles per permission.
- Each of these requirements can be satisfied by policies.
- Note, that the actual assignment of roles to users is outside the scope of the PDP.

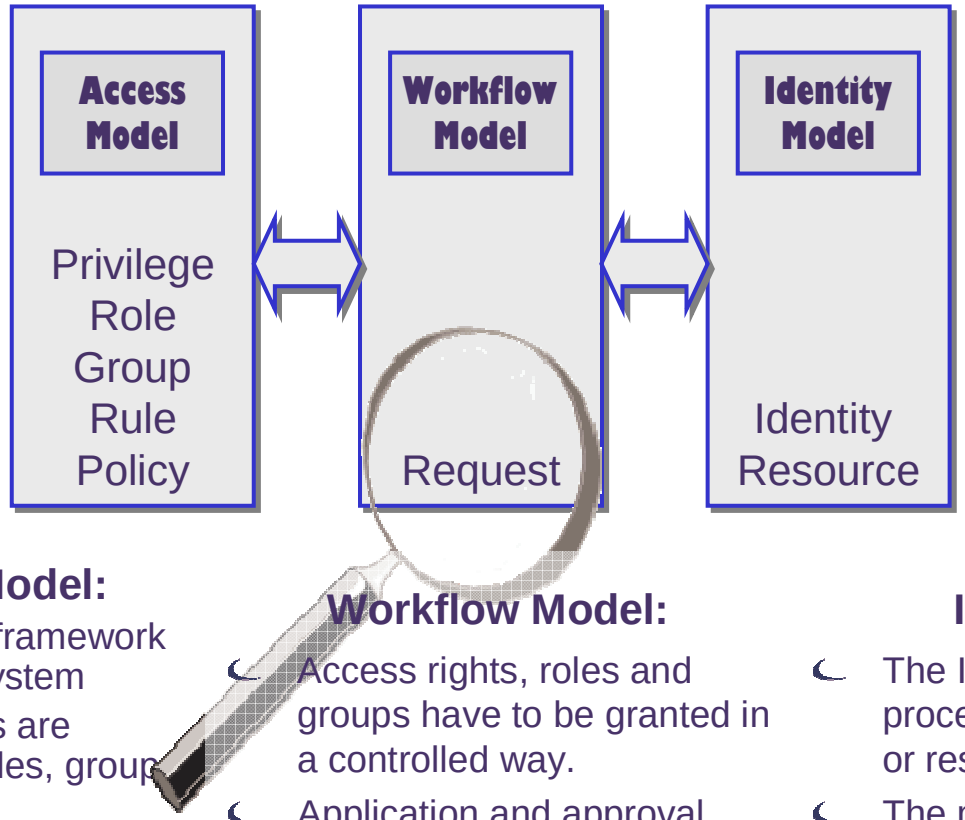
Layers of processes

how to include generic processes into a process model.



IAM Processes

Gartner Group defines three groups of IAM processes ….



Access Model:

- Describes a framework for an IAM system
- Major objects are privileges, roles, groups and policies.

Workflow Model:

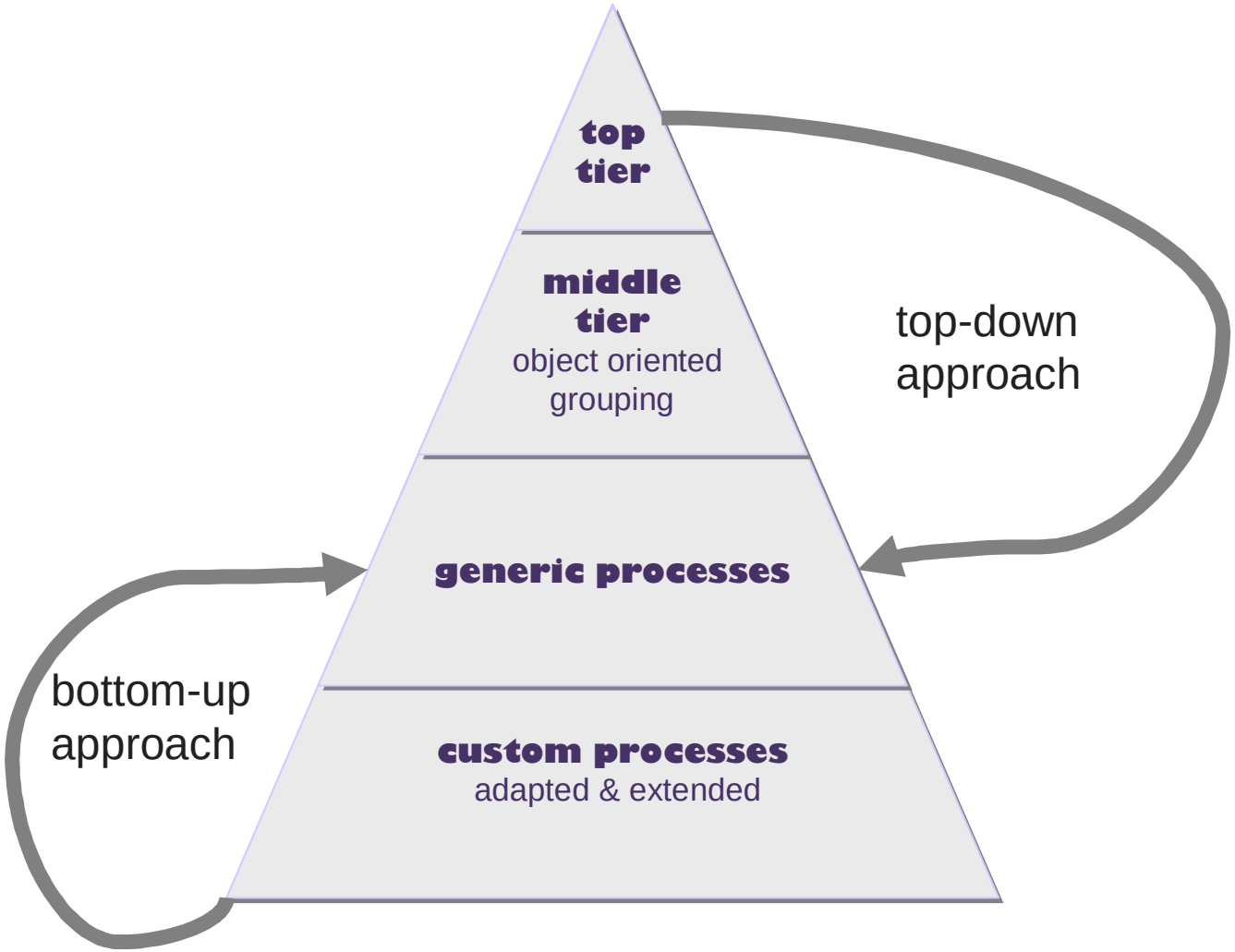
- Access rights, roles and groups have to be granted in a controlled way.
- Application and approval processes are located here.
- The main object is the request.

Identity Model:

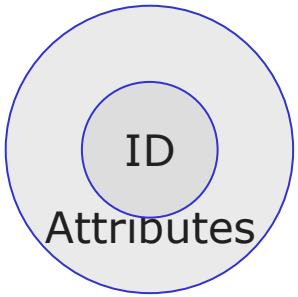
- The Identity Model contains all processes for specific identities or resources.
- The main objects are the identities and resources.
- IAM products implement many of the processes of this model.

Modelling approach

bottom-up- and top-down-approach lead to one generic model



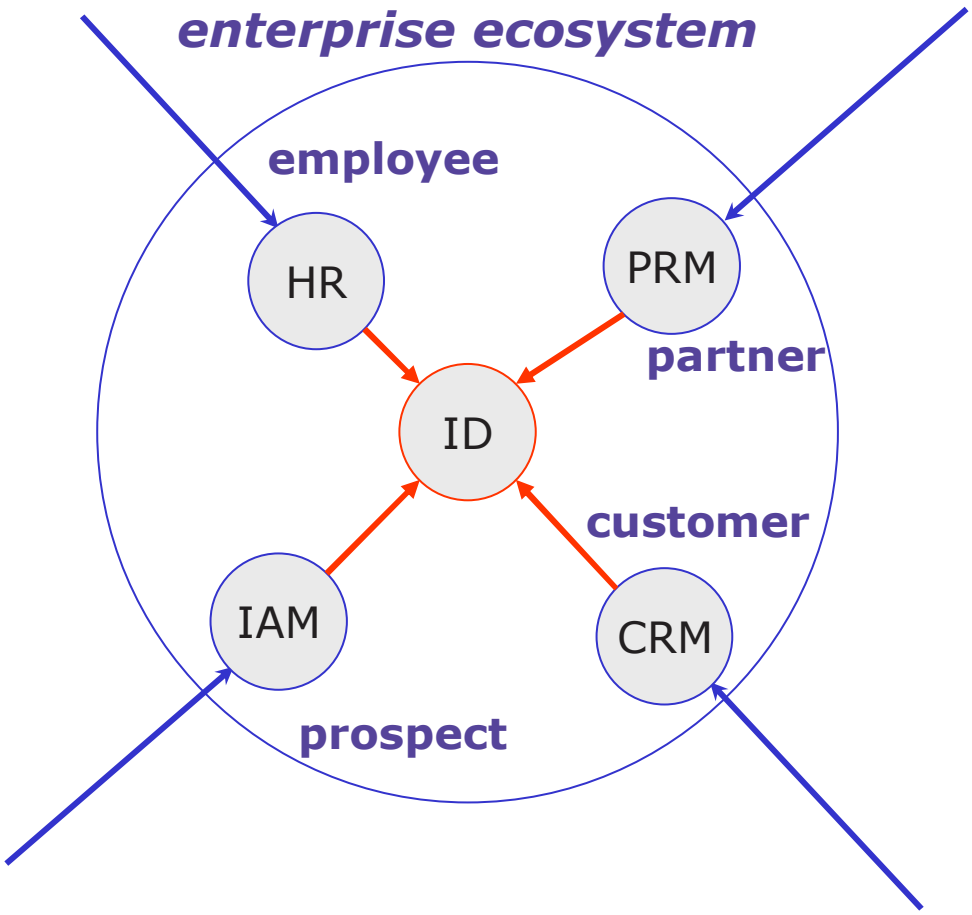
The Identity
and its “less rich” sibling the digital identity



- Identity is the fundamental concept of identity management
- In philosophy Identity is the sameness of two things.
- In object-oriented programming Identity is a property of objects that allows the objects to be distinguished from each other.
- But in Identity Management …
  - “*We usually speak of identity in the singular, but in fact subjects have multiple identities*.”
  - “*These multiple identities or personas, as they are sometimes called, …*”.
- The sum of all these Personas makes up the identity.
- In turn personas are to be understood as its projection to the space of information demand in a specific context.
- Biometrics ties the digital identity to the real world physical identity.

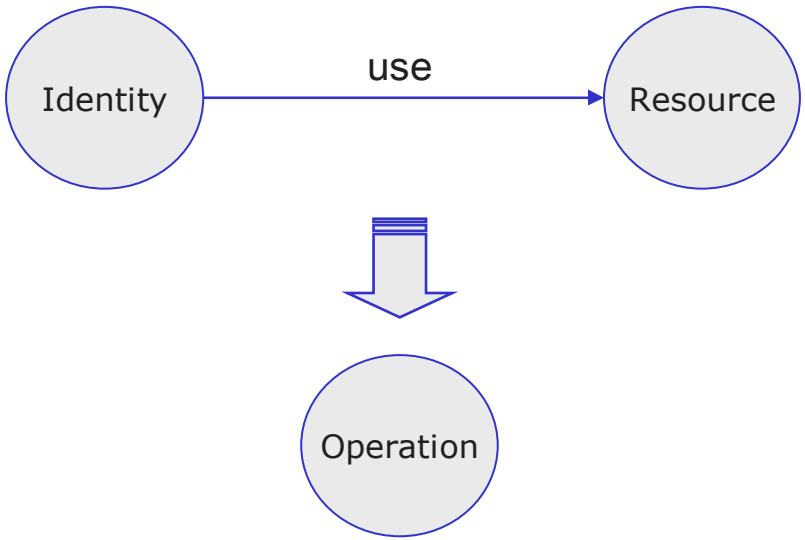
The central digital identity

whenever an individual enters the enterprise ecosystem first time …



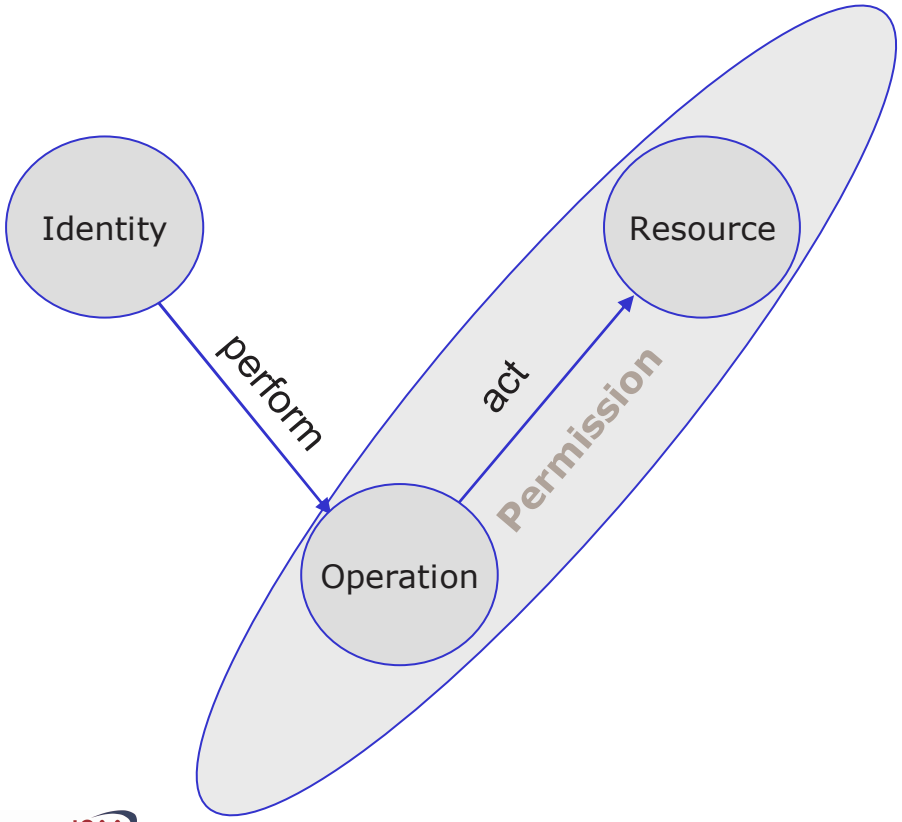
- Its digital identity is created whenever an individual enters the enterprise ecosystem 1st time.
- Regardless if it is a user or not
- Being a *user* represents a class of roles already
- The digital identity is the individuals digital sibling.
- Its lifetime is determined by the lifetime of the enterprises interest.
- The digital identity is global and unique
- It carries the minimal identifying attributes.

The Identity uses a Resource



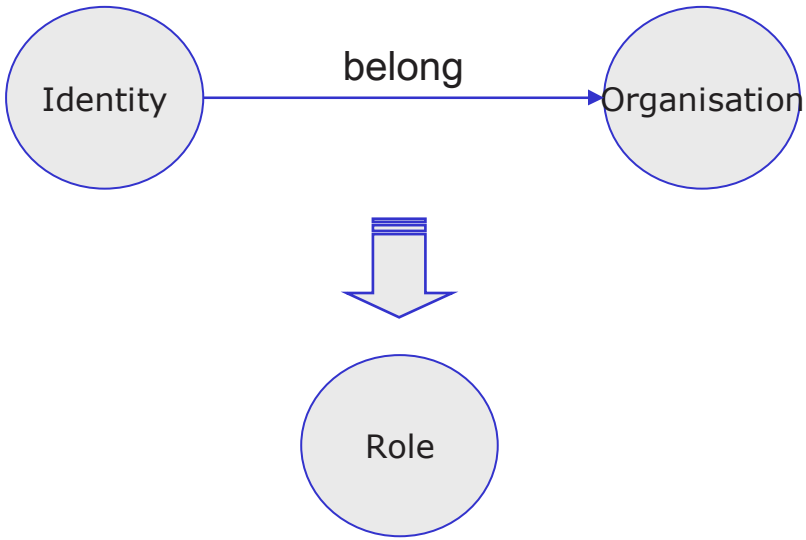
- Identities are often tied to resources
- They „use“ resources
- They do so by performing operations
- This relations may carry attributes
- It turns to a derived object: the user.

Permission = Operations on Resources



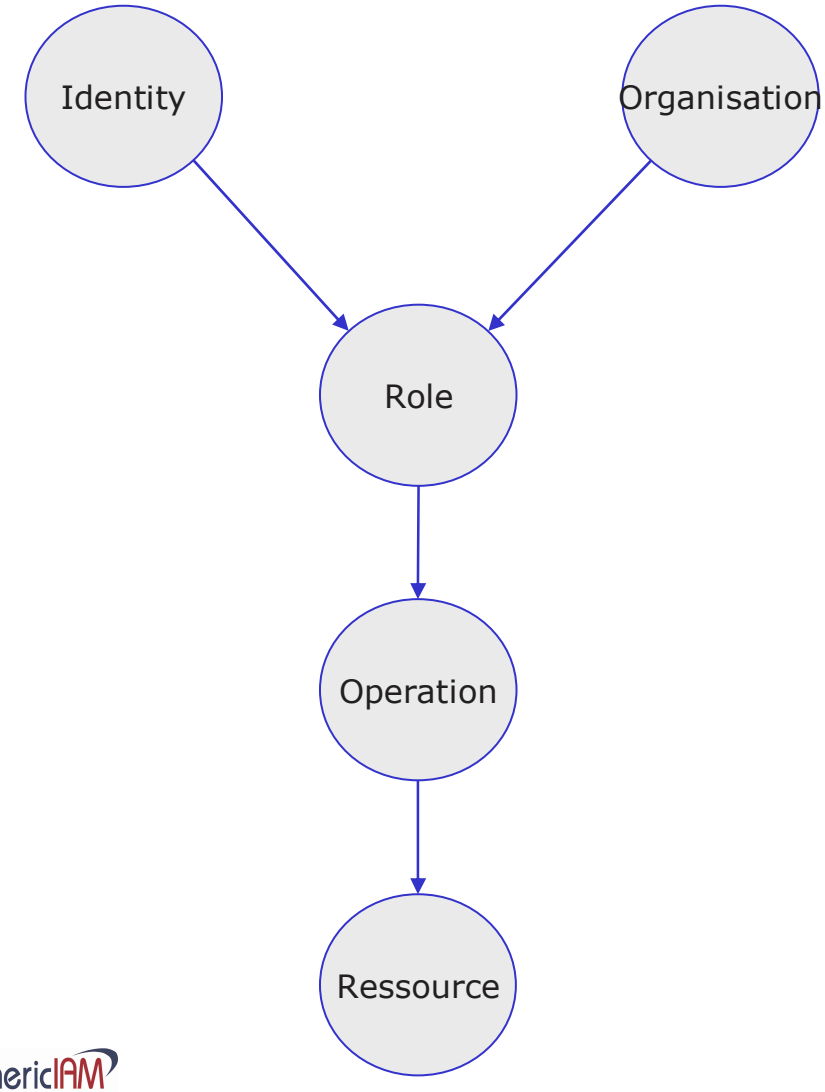
- The Identity performs an operation
- The operation acts on the resource
- Operations on resources (objects) may be labelled with “permissions”.
- Permissions are elementary
  - They are simple by definition
  - There may be a large number
  - There is a limited set of permissions

The Identity belongs to an organisation



- The Identity has a relationship to an organisation
- There are many specialisations to this relationship
- There might be more than on relationship
- This relationship may carry attributes
- It turns to a derived object: the role.

The Identity belongs to an organisation



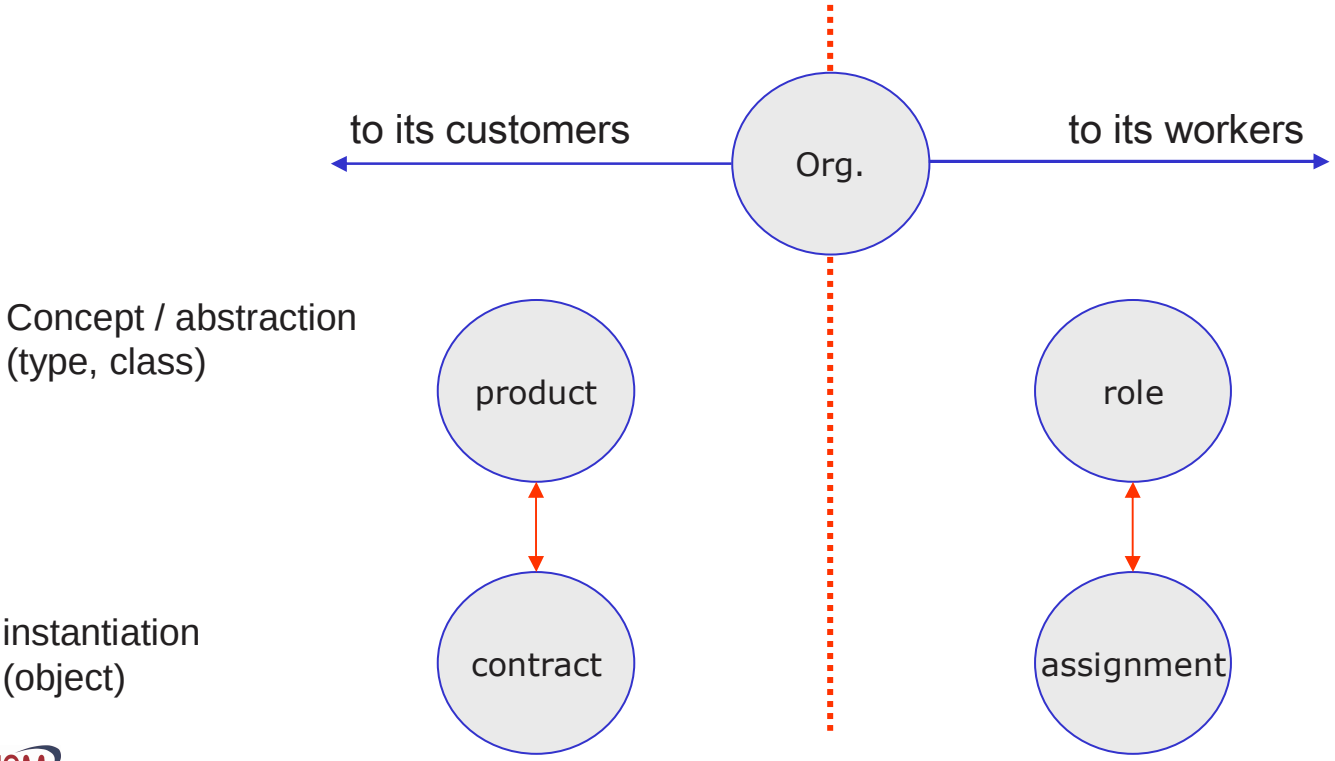
- The Identities role in an organisation …
- Performs operations on resources
- The role has a fine structure.
  - A contract defines the relationship
  - The roles define incarnation details

The role

- The role is an abstraction
- Like the „product“ abstracts the „contract“
- Hence the role relates to assignments like products to contracts.
- The privilege assignment looks similar to an employee contract.
- Both may in fact may be one “agreement”.
- They may as well be left separate.
- A customer may draw a privilege assignment as well.
- The (privilege) assignment and the contract may well be one agreement (collapse to one).

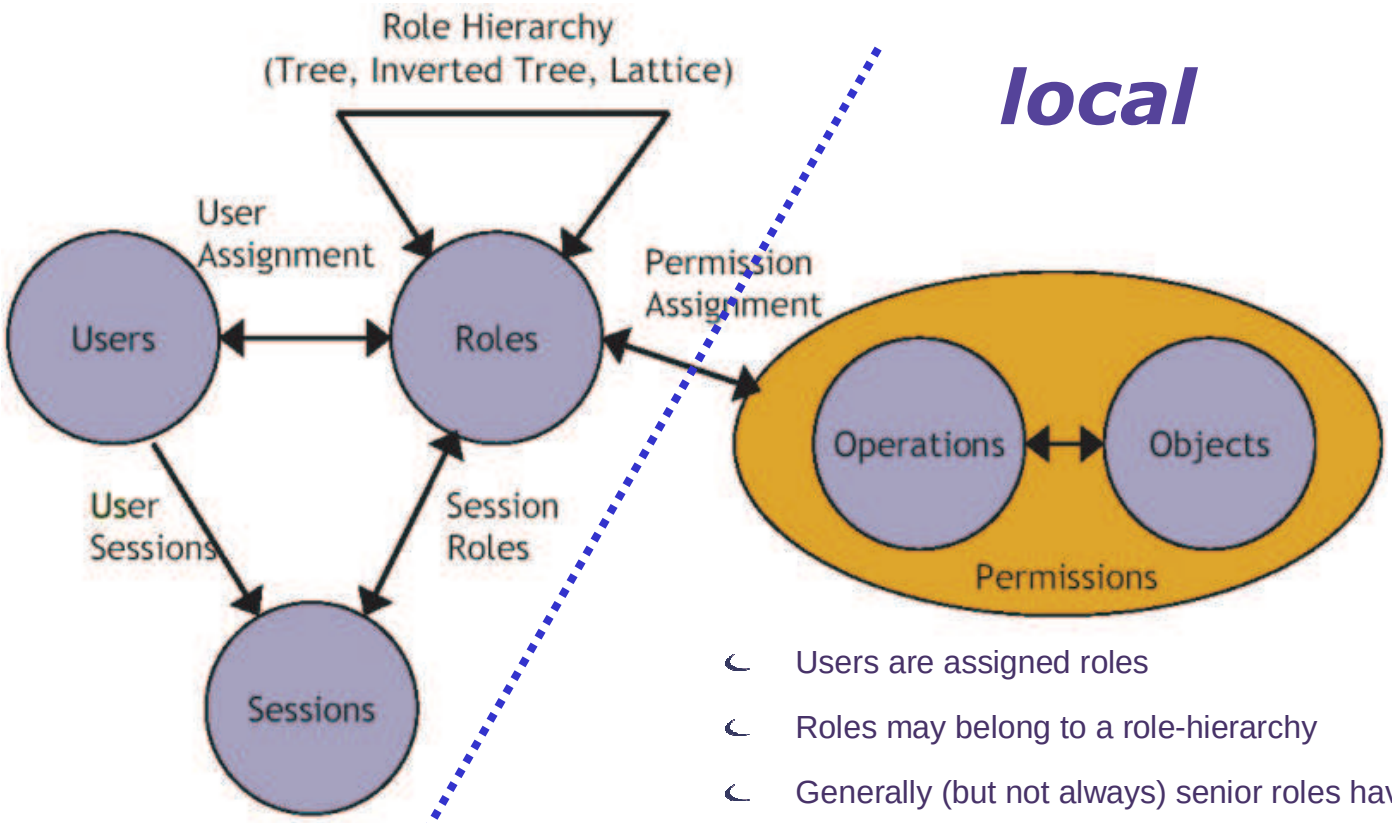
The concept of a role is an abstraction
like the product to the contract.

- The product generalises the contract
- The contract instantiates the concept of a product.
- The role generalises the (privilege) assignment.
- The (privilege) assignment instantiates the concept of a role.



Central vs. Local

IDs & roles are central by nature, while permissions are local



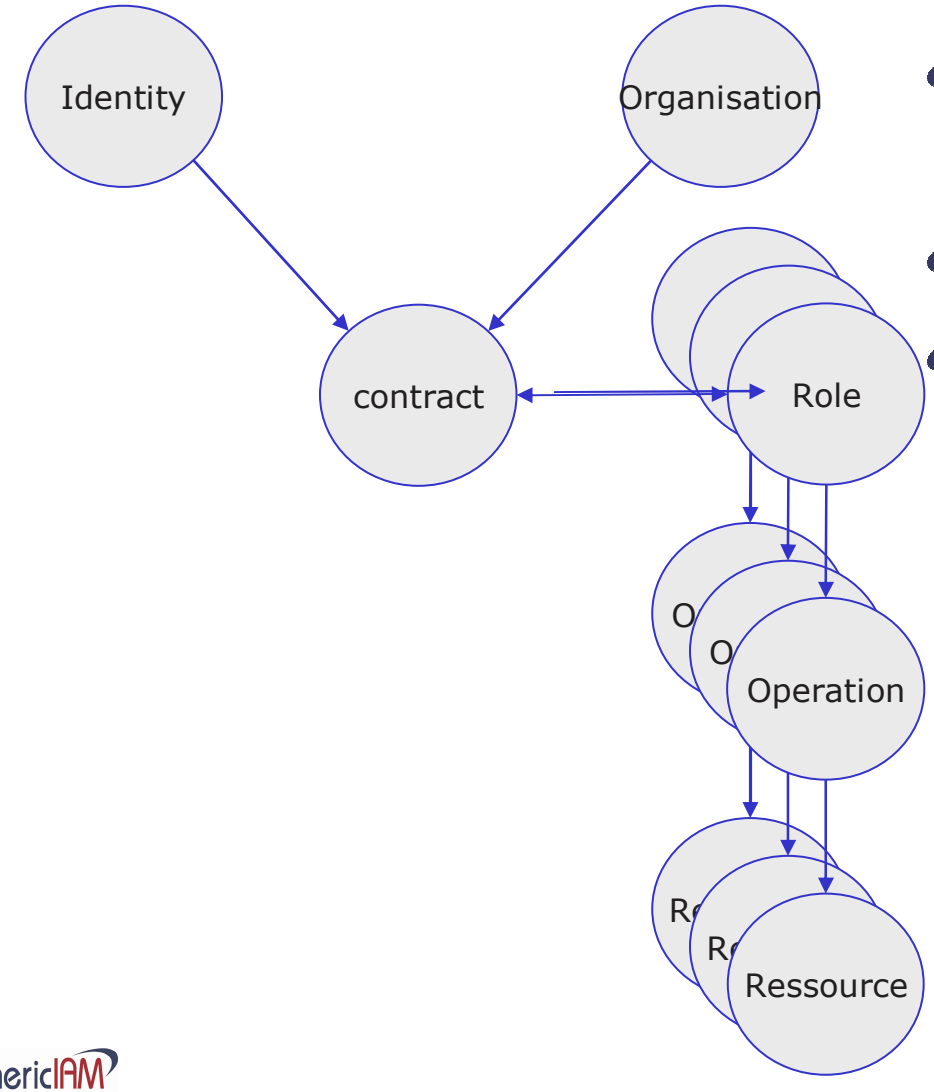
local

central

- Users are assigned roles
- Roles may belong to a role-hierarchy
- Generally (but not always) senior roles have all permissions assigned to junior roles
- Permissions are operations on objects.
- Permissions can be assigned + (additional) or - (subtractive)
- Roles Standard for Role Based Access Control

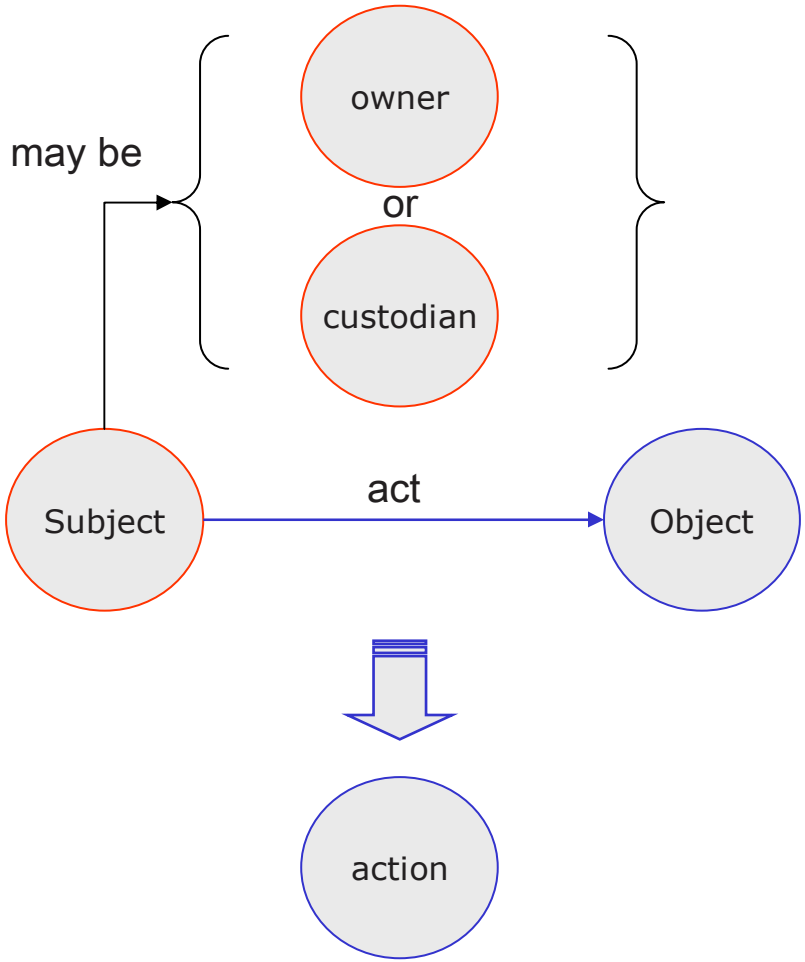
Source: Ferraiolo, Sandhu, Gavrila: A Proposed Standard for Role-Based Access Control, 2000.

Relationships are fixed in contracts



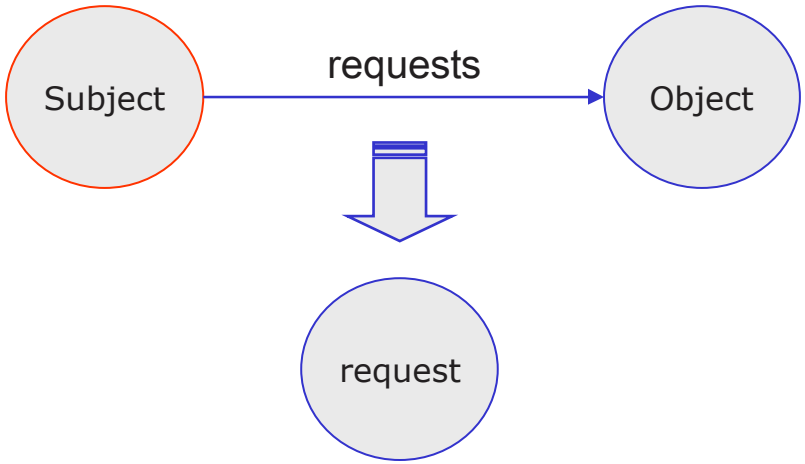
- The Identities role in an organisation …
- Performs operations on resources
- The role has a fine structure.
  - A contract defines the relationship
  - The roles define incarnation details
  - “the contract is expressed by several roles”

Subjects are acting on objects

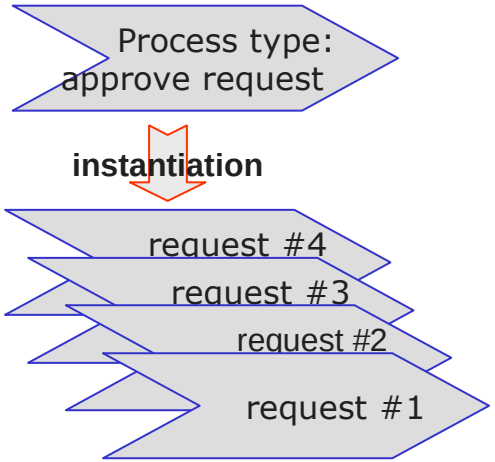


- In workflows subjects (actors) act on objects
- Subject may be an owner or a custodian
- Owners are responsible
- Custodians act on behalf of owners
- Owners delegate to custodians
- Subject act or react
- Their action triggers an event
- Reactions often are approvals

Request & approval

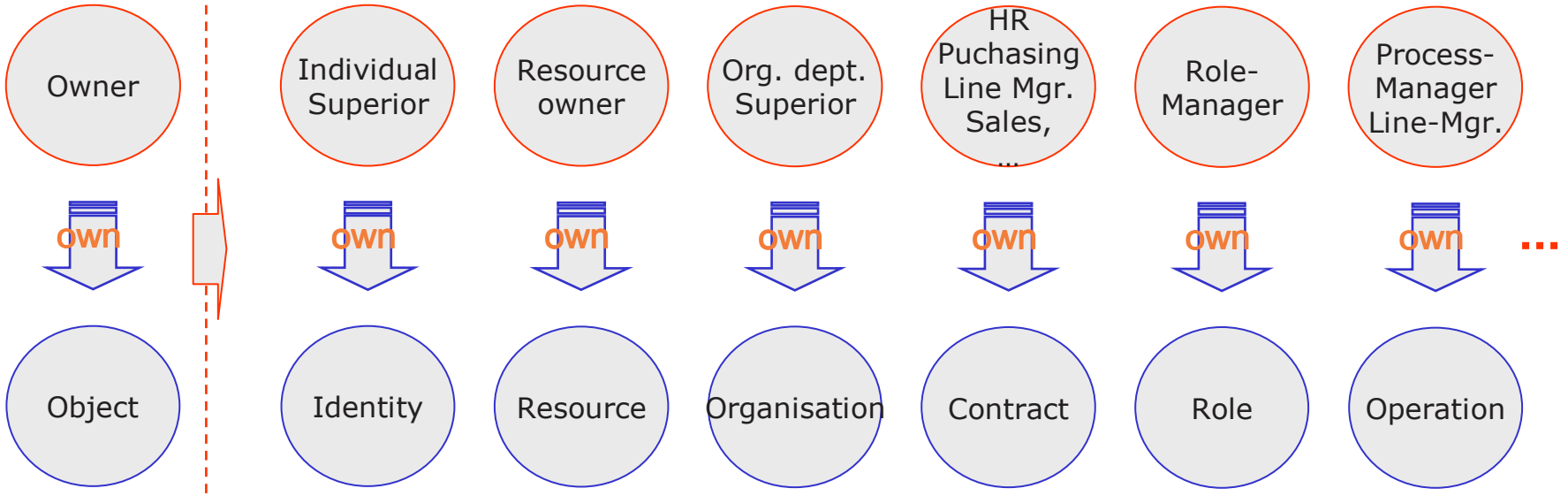


- The request is a transient object.
- It can be understood as the instantiation of a process type.
- The request is created by an event.
  - E.g. when a subject requests access to an object.
  - Or when time has come to re-validate a role / privilege.

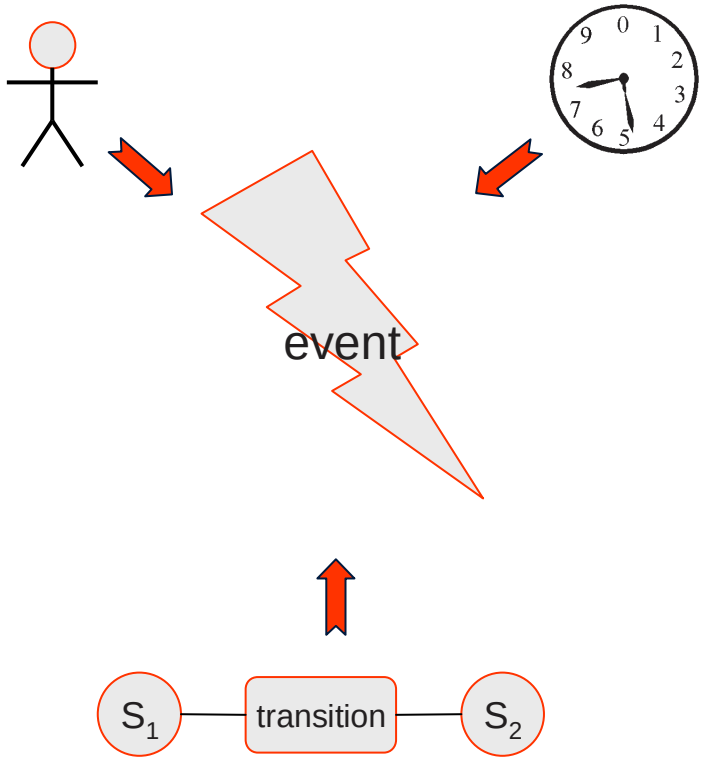


Every object has an owner

- Each object as one owner
- The owner is responsible for the object
- The owner may delegate object management to a custodian.
- The owner may temporarily transfer ownership (full responsibility) to delegate.
- Owners differ considerably from one organisation to another
- This apparent complexity is a result of customising a simple model



events

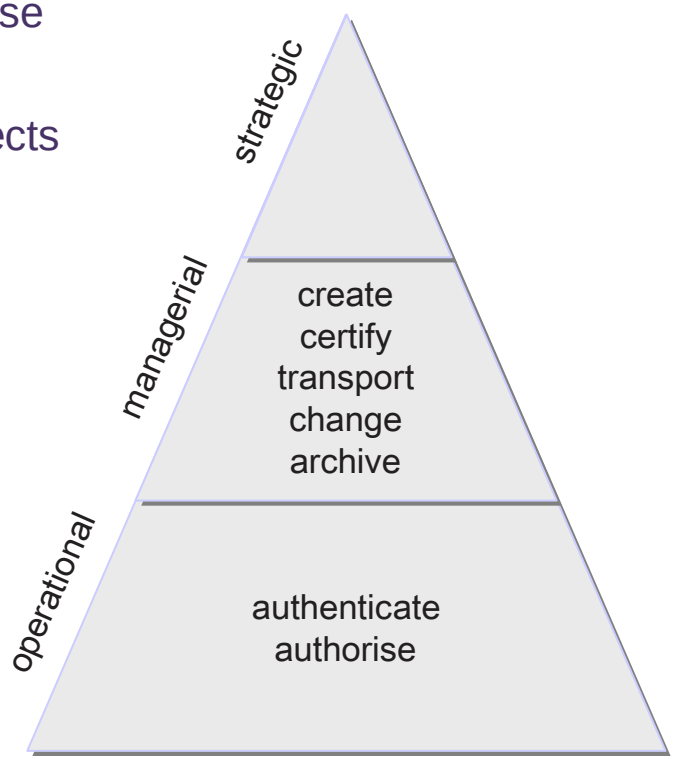


- There are events …
  - Created by an subject
  - Time triggered events
- State transitions fire events

Processes of the Identity Management

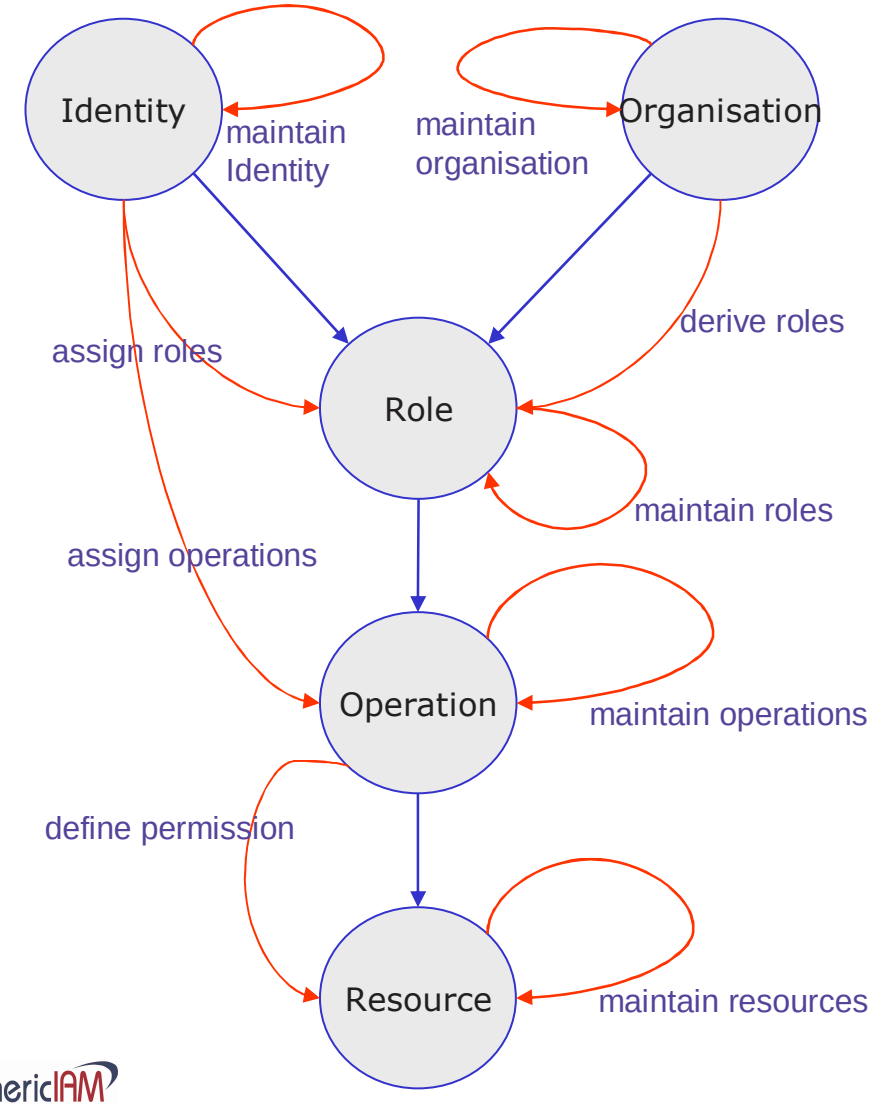
The Processes of the Identity Management may be grouped ...

- into operational, managerial and change
  - operational: identify, authenticate and authorise
  - managerial: administer digital Identities
  - Change: changing the implementation of objects
- into essential and physical
  - essential: administer and use
  - physical: integrate, transport, transform and “provision”
- into existence, certificate and context
  - create, read, change, delete
  - certify, revoke
  - assign, change, remove roles and privileges



➔ each classification has its specific value.

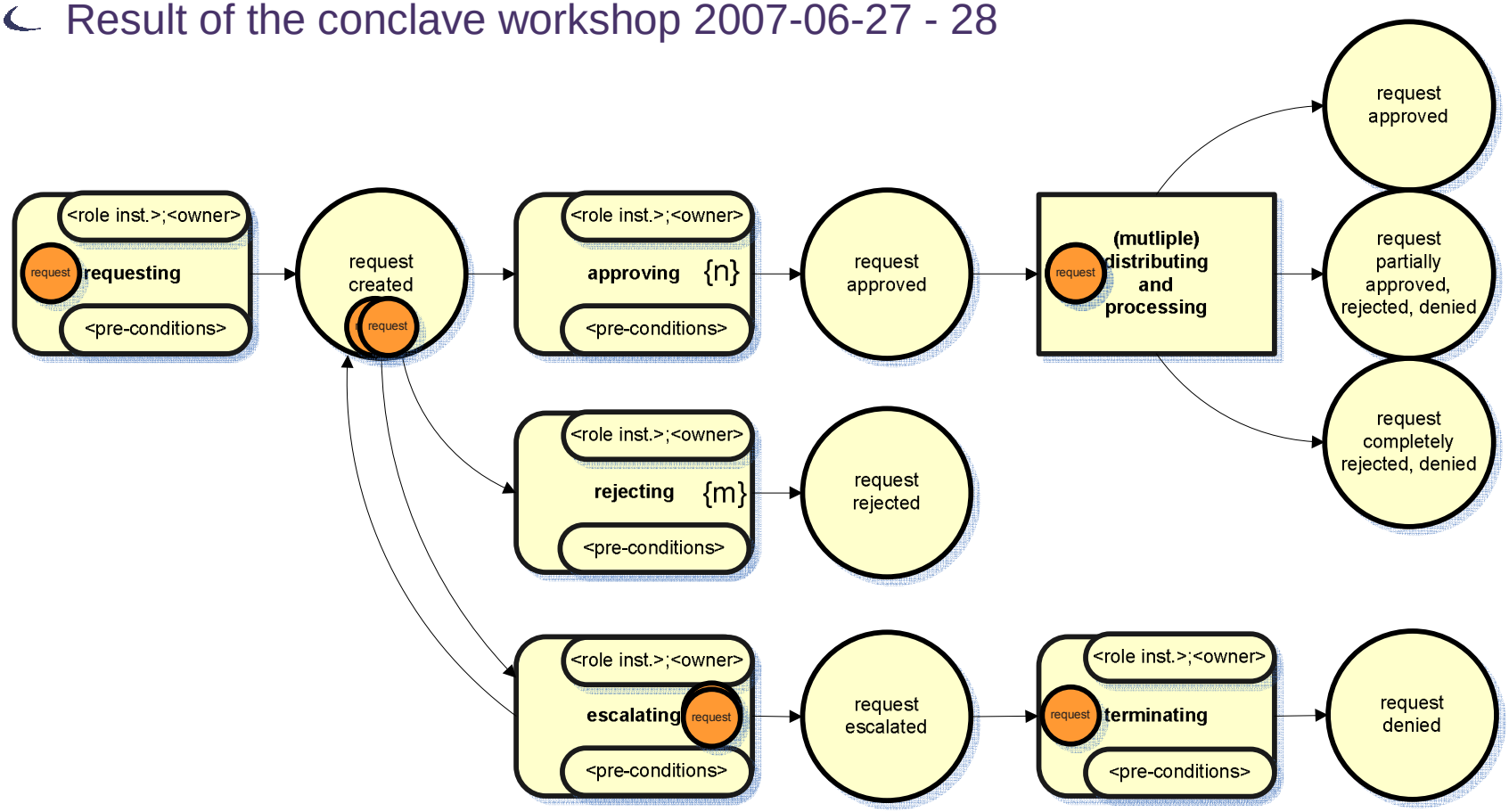
Elementary actions – changes on objects



- Processes consist of >= 1 activities.
- They are triggered by an event.
- They lead to a meaningful result to a subject.
- Process types (the class or definition) and process instantiations (incarnation, actual).
- Operational processes and managerial processes.
- Operational processes: *identification*, *authentication* and *authorisation*.
- The managerial:
  - *administrative processes*,
  - *audit processes* and
  - *change processes*.
- The administrative processes represent the "lions share" of all IAM processes.
- Its most prominent representative is the "*request & approval process*".

Approve request
generic process example using petri nets

- Result of the conclave workshop 2007-06-27 - 28



Questions - comments – suggestions?

Caution Appendix

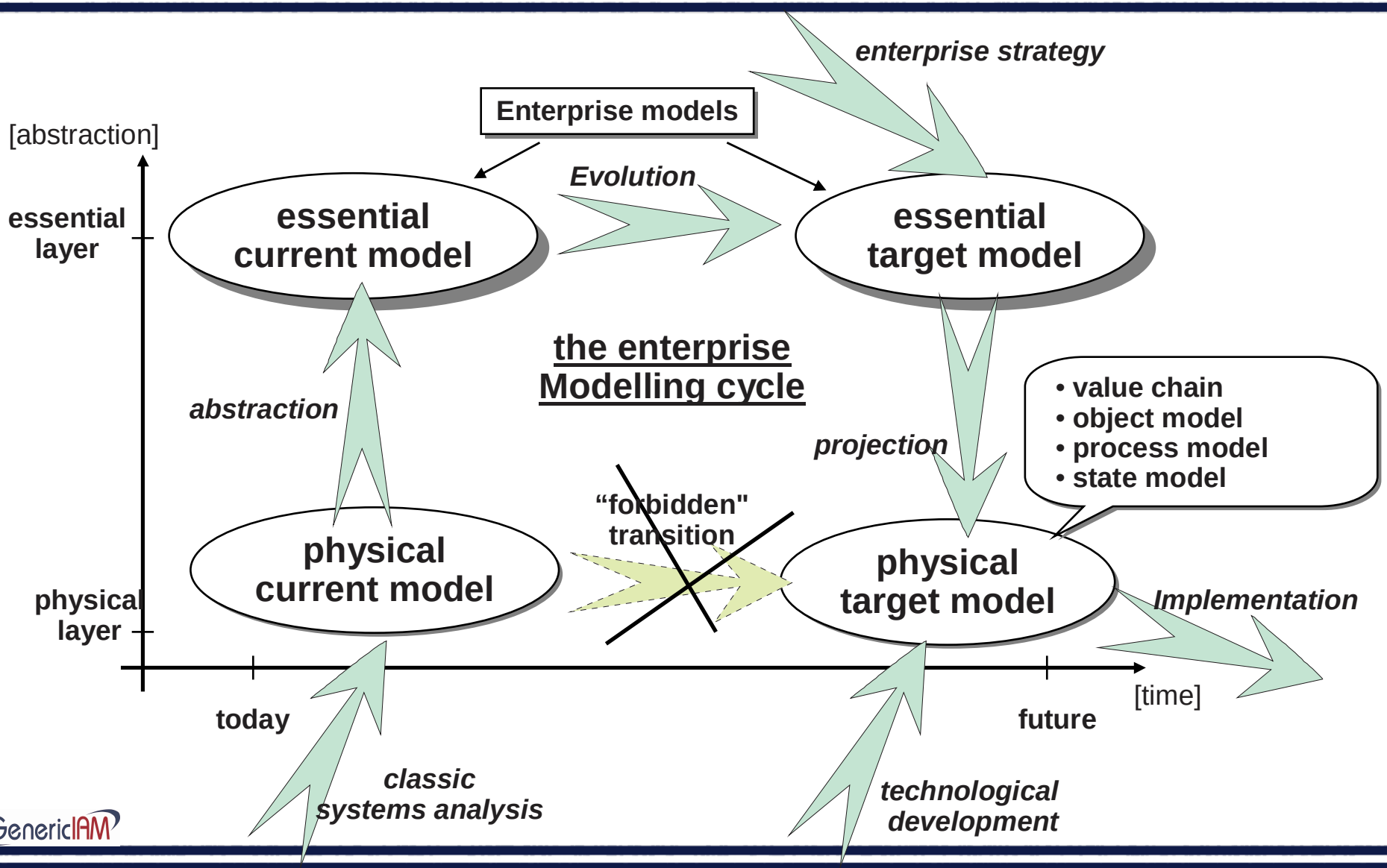
Here the notorious back-up-slides follow ...

Modelling process

In a four step Process to the target implementation model

- McMenamin and Palmer 1984 recommend to start a **four-step** Specification process with the analysis of the source model :
  - Analysis of the current systems; creating a model of the current implementation of the system.
  - Analysis of the fundamental concepts of this Implementation: creating a model of the essence of the current system. It will be abstracted from all implementation specific properties des (perfect technology).
  - Deriving the requirements to the new system: creating a model of the essence of the target system. This model describes the requirements and is not affected by any implementation considerations.
  - Designing the target system: creating a model of the implementation model of the target system.
- The requirements specification is limited to the 3rd step.

The modelling cycle
finding the essence removes implementation artefacts

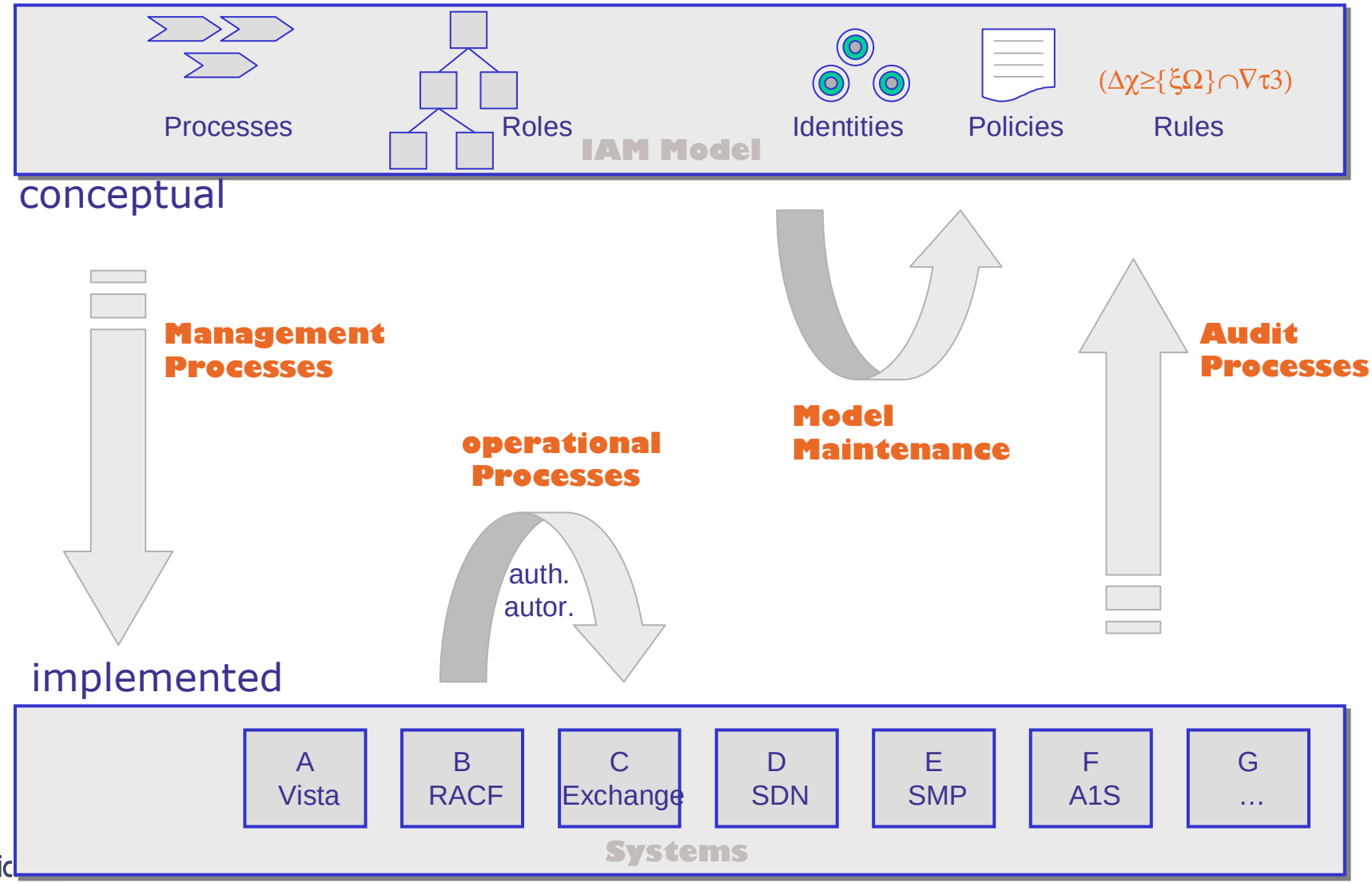


essential modelling
avoiding technical „folklore“ through perfect technology

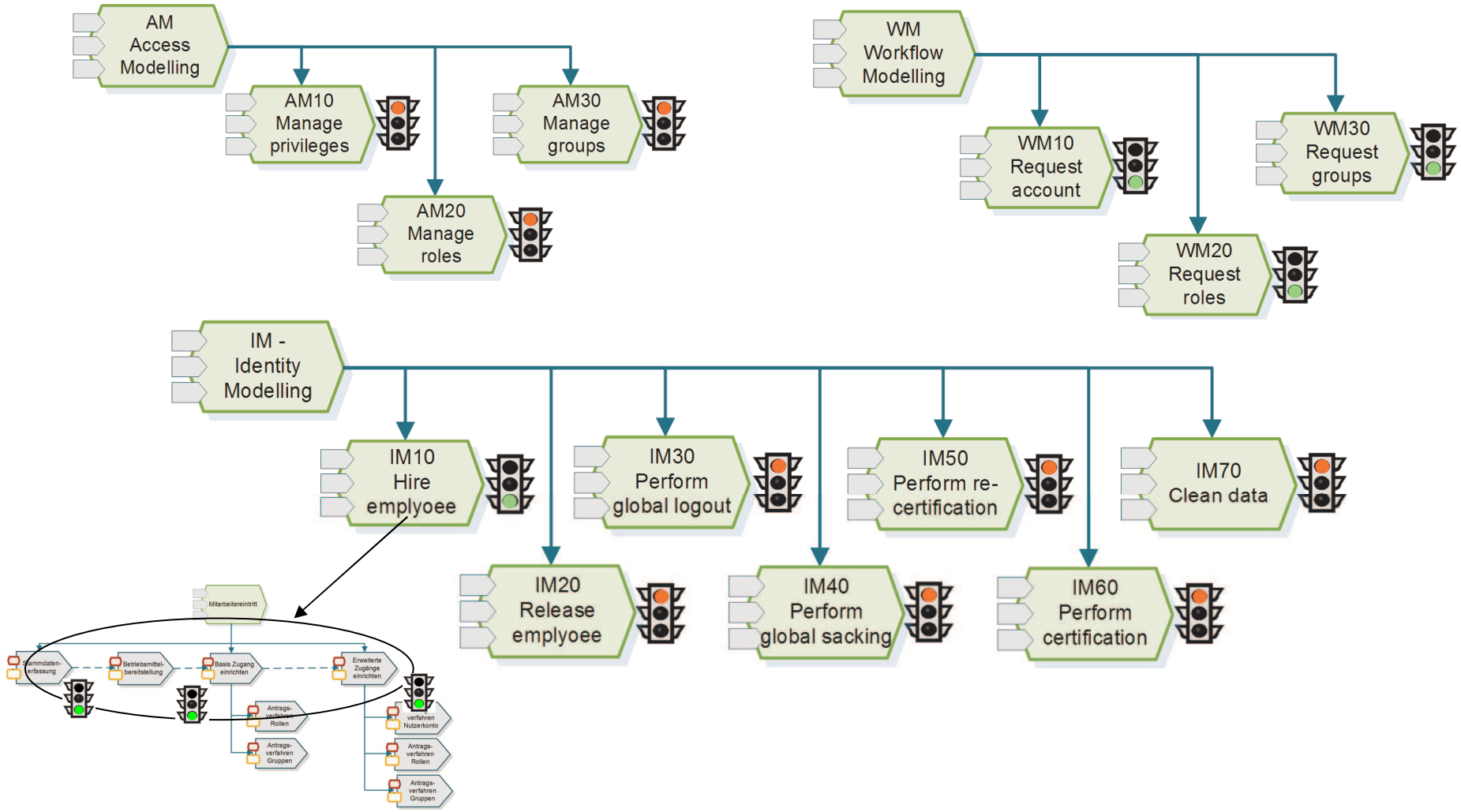
- McMenamin and Palmer require the existence of perfect technology for the System to be modelled.
  - in the **internal** neither errors nor processing- or waiting times occur.
  - check, translation und transport processes are absent there.
  - the **system context** is considered as imperfect.
  - at the **System border** there is a physical ring of these check, translation und transport processes .
- Essential Processes are triggered by external of by time events.
- Fundamental essential processes deliver an external result.
- Administrative essential processes store a result internally for a fundamental essential process.
- Essential Processes communicate asynchronously via essential stores – they are time decoupled.

Common IAM-Ownership

A central responsibility ensures a seamless architecture



generic process candidates
Identified the bottom-up way



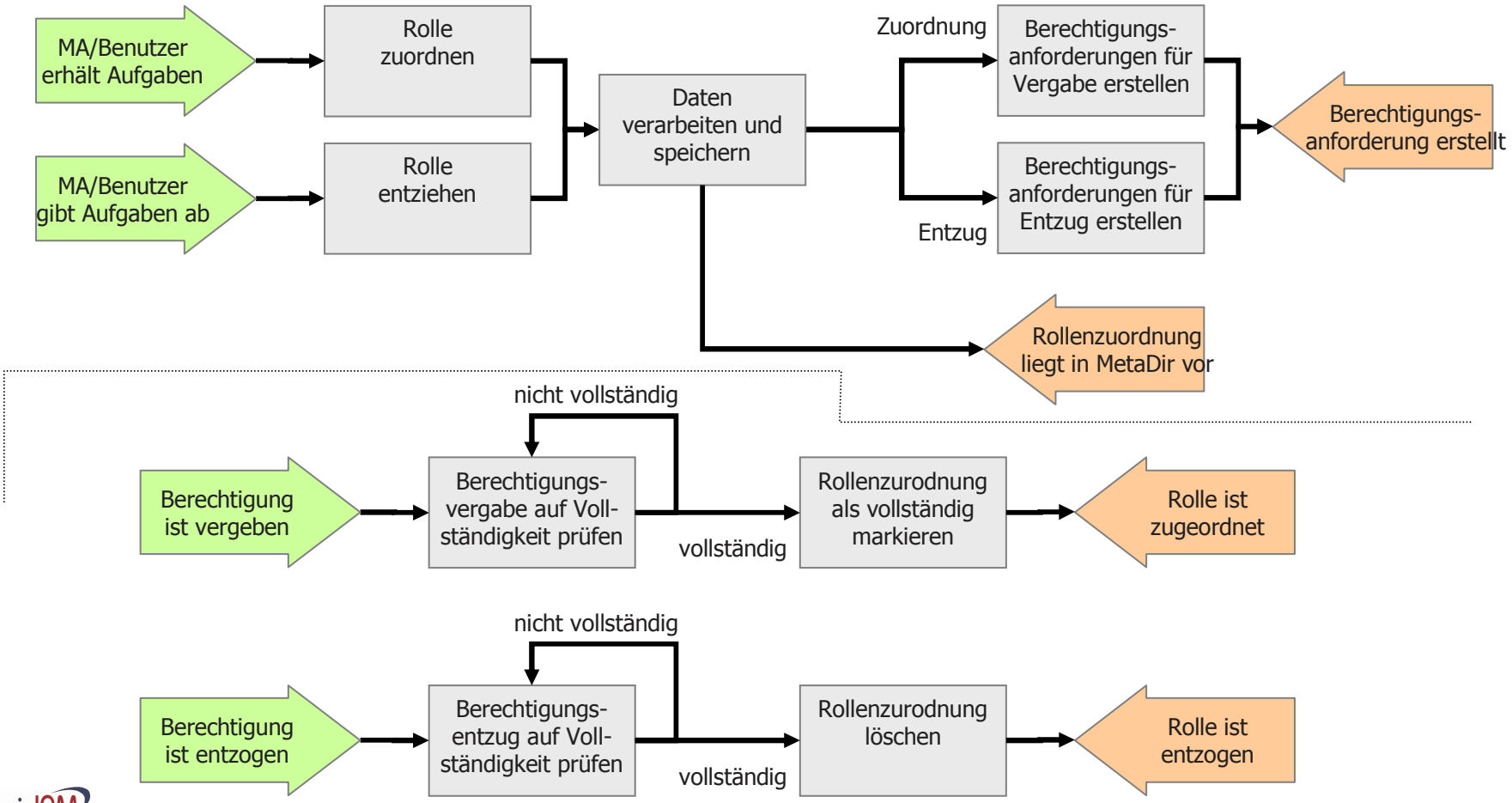
Validation of input against model

Bottom-up & top-down should meet somewhere …

- Example processes from input source have to be mapped against the proposed model:
  - Dekra
  - BMW
  - WestLB
  - DoubeSlash
  - ism
  - Others …

Detail of the Dekra Processes

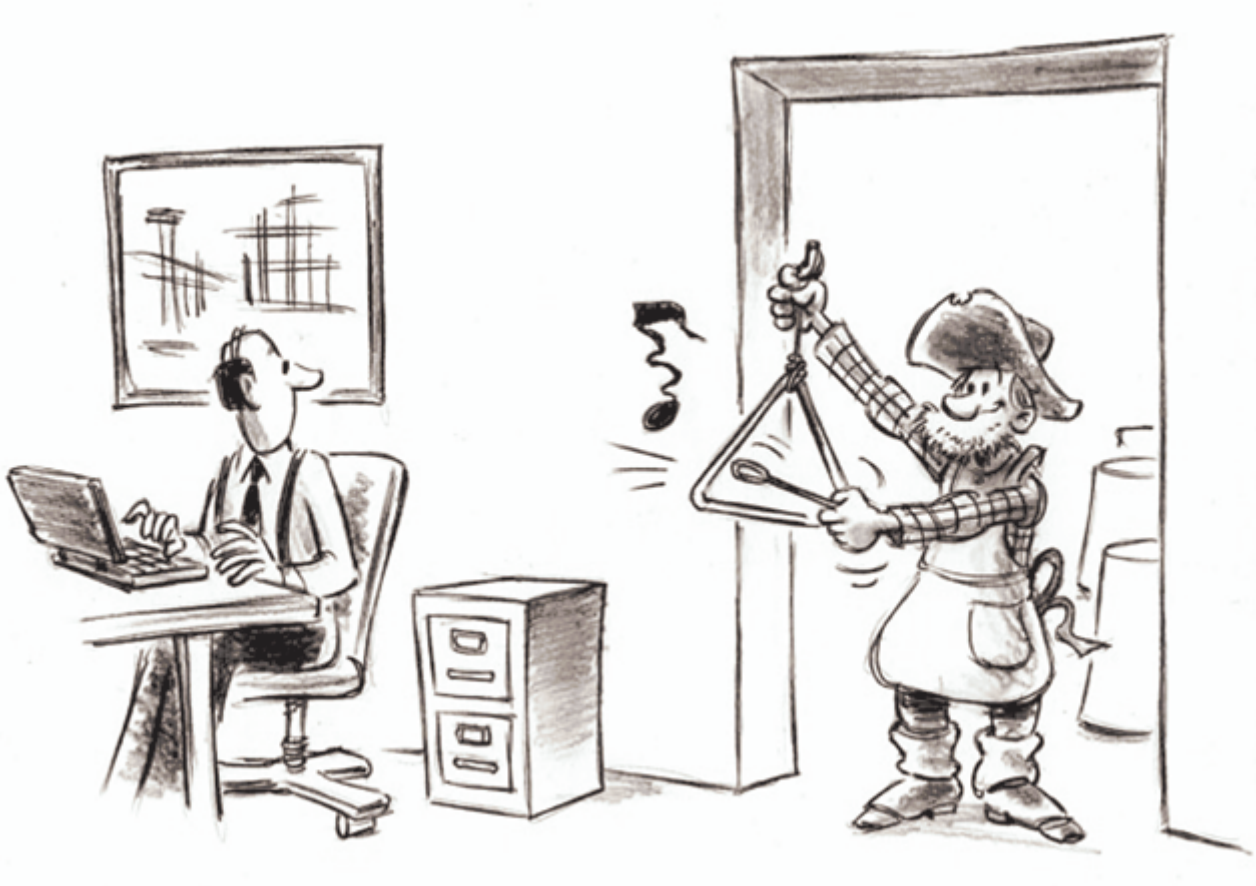
Role assign / remove



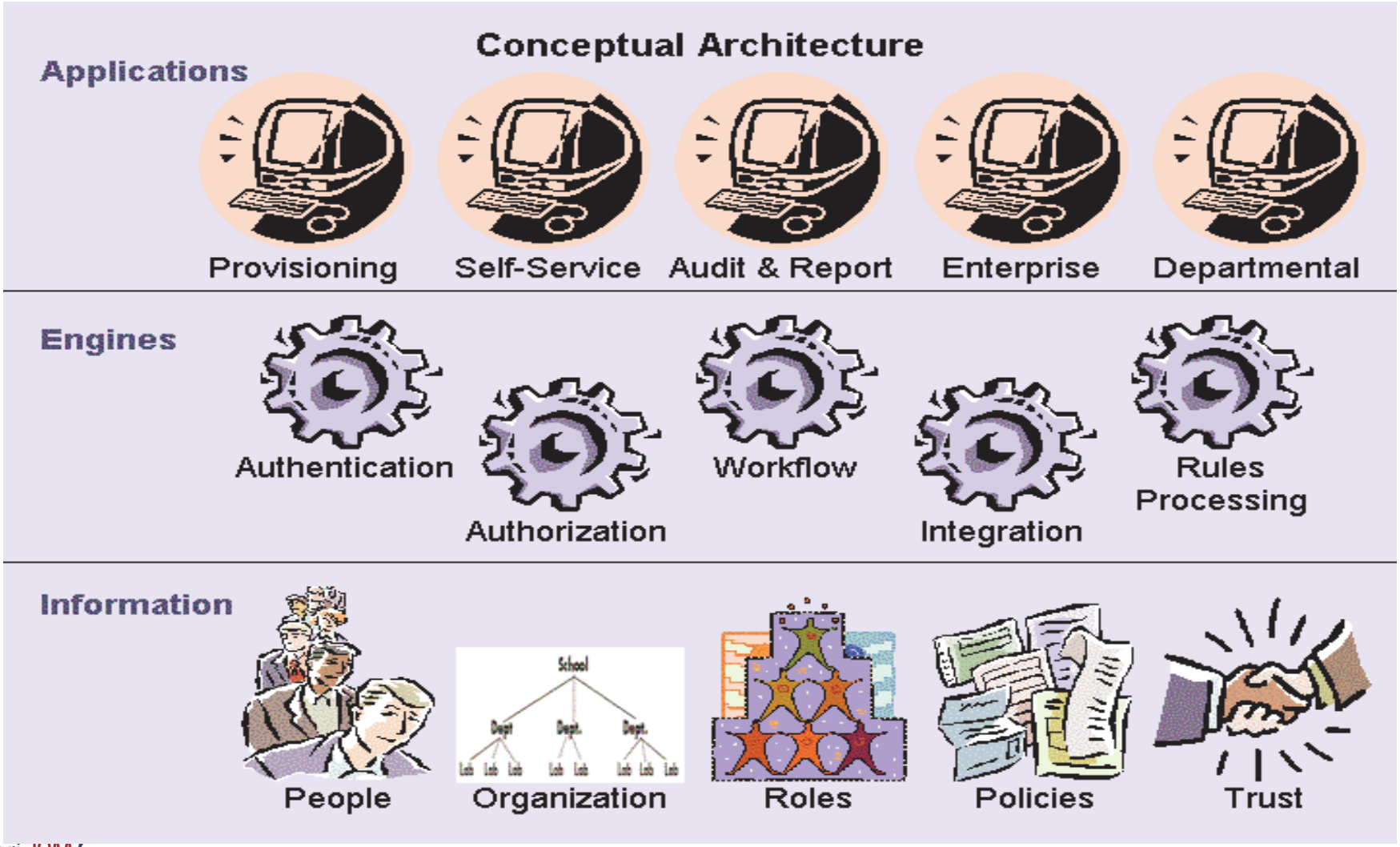
Reference Model - examples

- Supply Chain Operations Reference Model (SCOR-Modell)
- IDS Scheer führt Value Reference Model (VRM) für unternehmensweiten Support über die 'ARIS Platform' ein

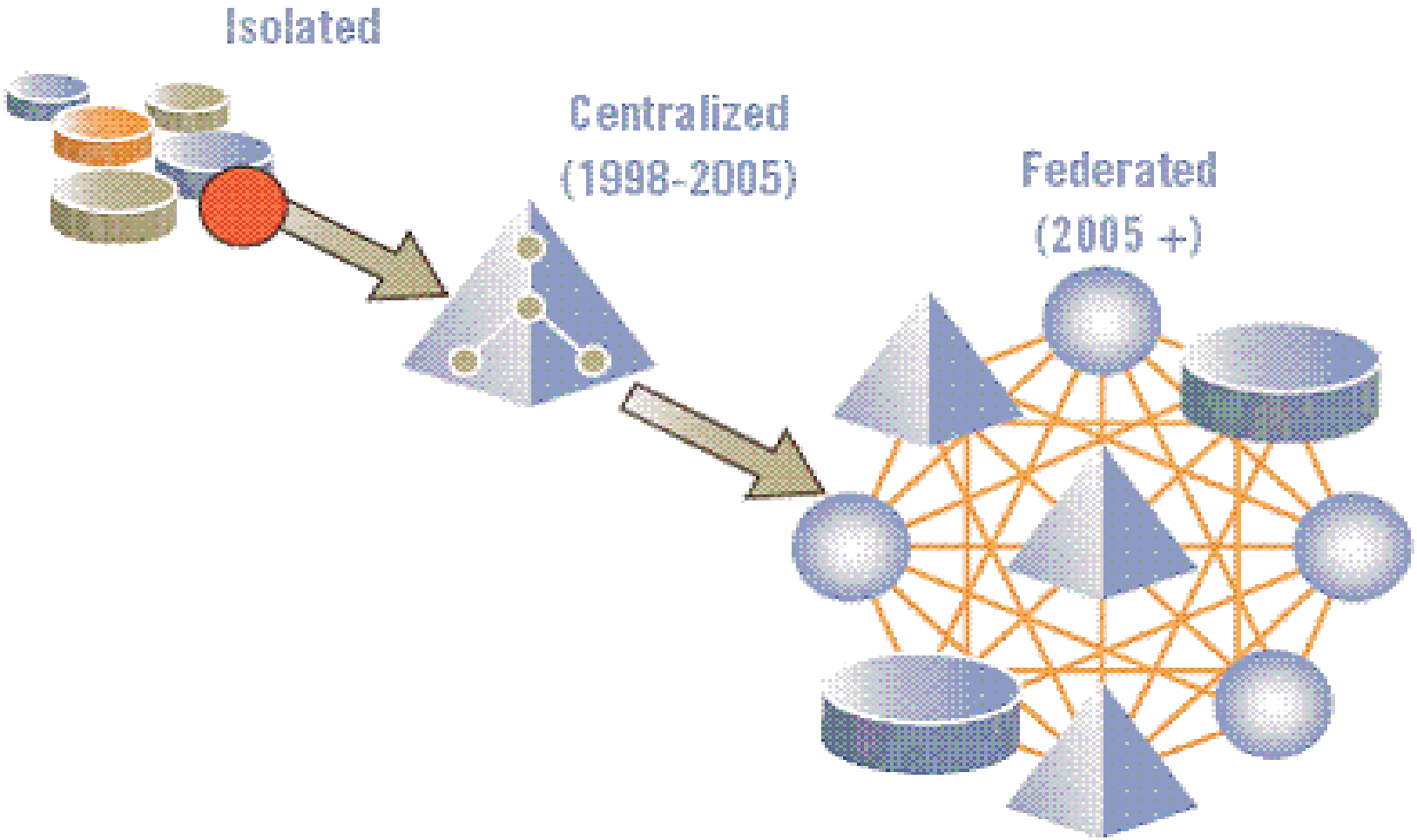
Lunch break



SOA – The Identity layer



Evolution of identity centralisation



Identity- & related Managements

- Identity Management
  - Access Management
  - Personalisation
  - Compliance Management
- Identity: to find out who you are
- Trust is being built by time
- IdM generations
  - Identity 1.0 = Silo
  - Identity 1.5 = Federation
  - Identity 2.0 = user centric

The Vaau approach