



Industrial Automation
Automation Industrielle
Industrielle Automation



5

SCADA

Operator Interface

interface homme machine

Mensch-Maschine Kommunikation

Prof. Dr. H. Kirrmann

ABB Research Center, Baden, Switzerland

Control room



Two human interfaces: old style mimic board (behind) and screens (front)

SCADA functionality

Data acquisition and display

- store binary & analog data into process data base

Alarm & Events

- record important changes and operator actions

History data base

- keep a record of the process values

Measurand processing

- calculate derived values (limit supervision, trending)

Logging & reporting

Human Machine Interface (HMI):

- graphical object state presentation, lists, reports

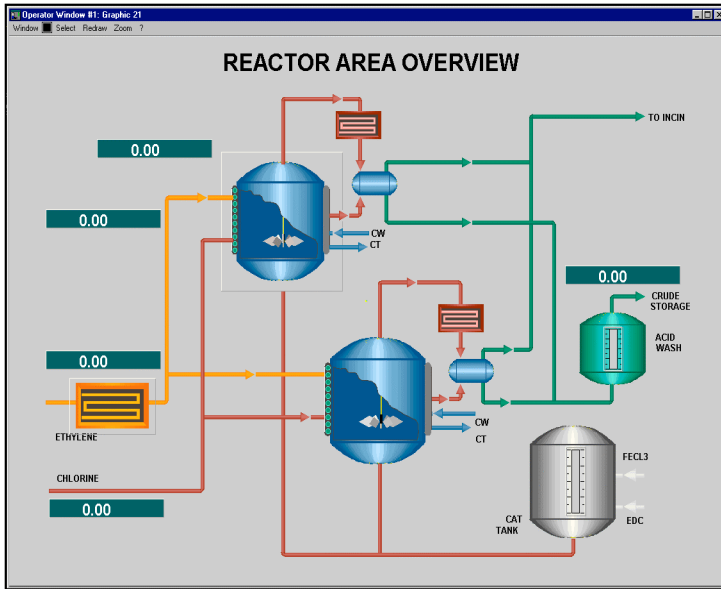
Operator Command handling

- binary commands, set points

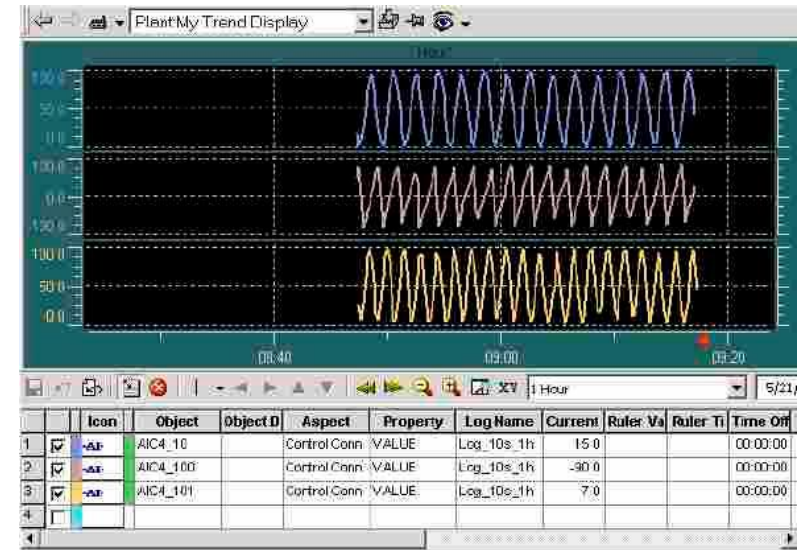
- recipes, batches, scripts (command procedures)

Interfacing to planning & analysis functions: CMMS, ...

Operator workplace: three main functions



current state



alarms and events

trends and history

The TeMIPClient interface displays a list of alarms and events. The table below shows the data from the main window.

S	P	A	C	Event Type	Perceived Sever.	Probable Cause	Managed Object
[icon]	[icon]	[icon]	[icon]	ProcessingE...	Warning	StorageCap...	OPERATIO...
[icon]	[icon]	[icon]	[icon]	ProcessingE...	Critical	StorageCap...	OPERATIO...
[icon]	[icon]	[icon]	[icon]	ProcessingE...	Critical	StorageCap...	OPERATIO...
[icon]	[icon]	[icon]	[icon]	Environment...	Major	CallE stablis...	OPERATIO...
[icon]	[icon]	[icon]	[icon]	Environment...	Critical	CallE stablis...	OPERATIO...
[icon]	[icon]	[icon]	[icon]	Environment...	Minor	CallE stablis...	DSL_SYSTE...
[icon]	[icon]	[icon]	[icon]	Environment...	Warning	AdapterError	DSL_SYSTE...
[icon]	[icon]	[icon]	[icon]	Environment...	Warning	CallE stablis...	DSL_SYSTE...
[icon]	[icon]	[icon]	[icon]	Environment...	Minor	CallE stablis...	OPERATIO...

Filtered Alarms (Total): 9 Filtered Alarms (New): 22

ObjectName	Monitored	Domain Name	Date	Message
dopin_Admin_...	[icon]	.dopin_Admin_Dom	11/14/2000 15:40:32	Disable DC_hds.hds_ooc2
hds.hds_ooc1	[icon]	.hds.dom1	11/14/2000 15:40:33	hds.hds_ooc2 successfully disabled
demo.radio_oc	[icon]	.demo.radio_system	11/14/2000 15:40:35	Disable DC_hds.hds_ooc2

Human-Machine Interface to Plant (HMI-P)

Representation of process state	<ul style="list-style-type: none">• Lamps, instruments, mimic boards• Screen, zoom, pan, standard presentation• Actualization of values in the windows• Display trends and alarms• Display maintenance messages
Protocol of the plant state	Recording process variables and events with time-stamp
Dialog with the operator	Text entry, Confirmation and Acknowledgments
Forwarding commands	Push-buttons, touch-screen or keyboard
Record all manipulations	Record all commands and especially critical operation (closing switches)
Mark objects	Lock objects and commands
Administration	Access rights, security levels
On-line help	Expert system, display of maintenance data and construction drawings, internet access

Human-Machine Interface to Engineering (HMI-E)

Configuration of the plant	<ul style="list-style-type: none">• Bind new devices• Assign names and addresses to devices• Program, download and debug devices
Screen and Keyboard layout	Picture elements, Picture variables, assignment of Variables to Functions
Defining command sequences	Command language
Protocol definition	What is an event and how should it be registered ?
Parameterize front-end devices	Set points, limits, coefficients
Diagnostic help	Recording of faulty situations, fault location, redundancy handling

Mainly used during engineering and commissioning phase, afterwards only for maintenance and modifications of the plant.

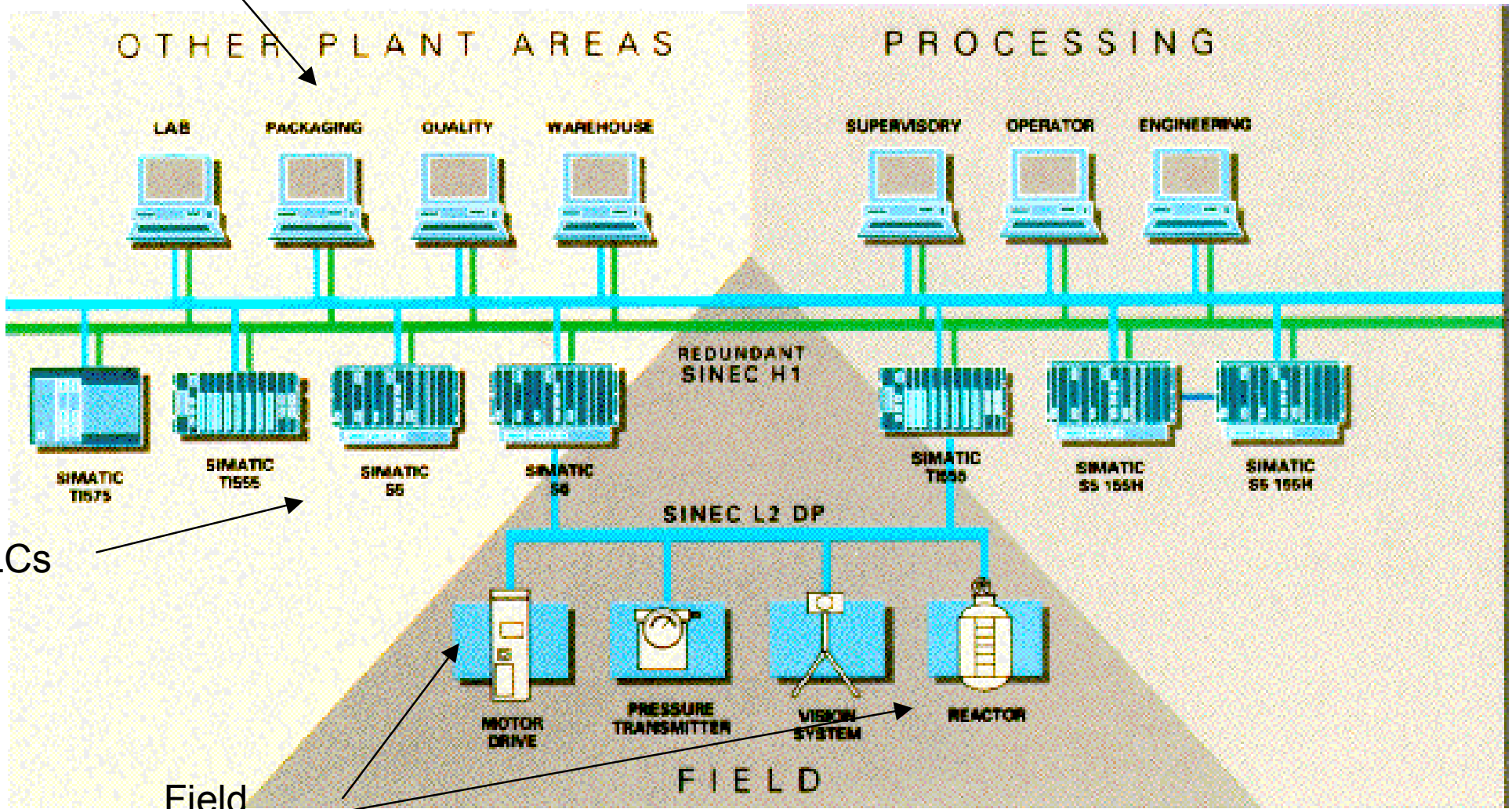
Used more often in flexible manufacturing and factory automation.

Local Operator Console (printing)



Example: Siemens

Workstations



PLCs

Field devices

www.aut.sea.siemens.com/pcs

Functions of the operator interface

- **Process Graphics**
- **Event/Alarm Manager**
- **Trends**
- **Historian**
- **Controller Integration**
- **Recipes**

Process graphics

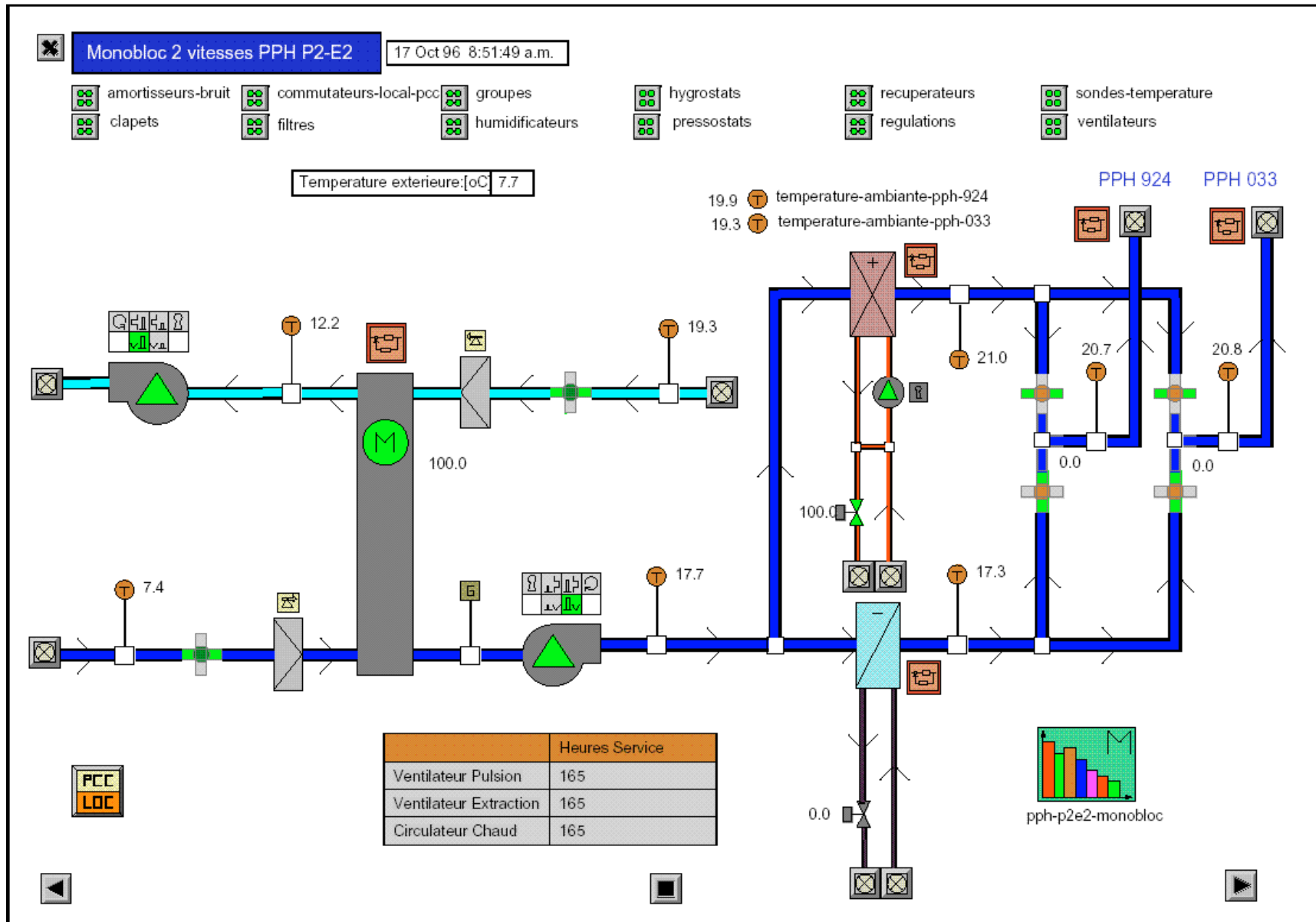
The screenshot displays a complex industrial process control system interface. At the top, there is a navigation bar with various area abbreviations (LAC, LAB, HKW, HNA, CPB, LAC, SYS) and a date/time display (20:02, 21.05.97). Below this is a menu bar (User, Event, Tag, Window, Service, Help) and a search/address field (GC-LIC-454). The main interface is divided into several windows:

- REACTOR AREA OVERVIEW:** A central process flow diagram showing two large blue reactor vessels. It includes various input streams (ETHYLENE, CHLORINE) and output streams (TO RECH, CRUDE STORAGE, ACID WASH, FECL3, CAT TANK, EBC). Flow rates are indicated as 0.00.
- Batch Scheduler:** A window showing a table of batch operations. The table has columns for Batch ID, Lot, Recipe, Status, and Batch Area Start Time. It lists various batch IDs and their corresponding recipes and statuses.
- Unit Overview:** A window showing a detailed view of a specific unit, including its status, operator, and various parameters. It includes a table with columns for Unit ID, Operator, Type, UnitType, Status, and Reserved.

Trends:

disappearance of custom HMI, increasing access over Windows (Internet Explorer), data entry by keyboard, touch screen, trackball (seldom mouse), buttons (hard-feel).

Example of Screen (EPFL air condition)



Example of Screen

SCHLAPPIN [1] - MicroSCADA

Steuerung Optionen Dateneingabe Listen Stationen Tools Hilfe

Betriebsprotokoll Mittwoch (W31)

Ereignisse von: 99-08-03 bis: 99-08-03 Intervall von: 99-08-03 bis: 99-08-03 Seite

Date	Time	Object Id	Object Text	Status
99-08-03	13:17:18.412	SCS Monitor 7	Benutzer: SCHLAPPIN	Login
	13:43:13.097	KW S GENERATOR REC	RM Hand ein	Geht
	13:43:28.660	KW S GENERATOR REC	RM Hand ein	Kommt
	13:43:34.688	KW S GENERATOR REC	RM Hand ein	Geht
	13:43:38.093	KW S GENERATOR REC	RM Hand ein	Kommt
	13:43:42.630	KW S GENERATOR REC	RM Hand ein	Geht
	13:43:45.854	KW S GENERATOR REC	RM Hand ein	Kommt
	13:46:20.897	KW S TURBINE DTL	DTL startbereit	Geändert
	13:46:31.062	KW S TURBINE DTL	DTL startbereit	Geändert
	13:49:47.404	KW S TURBINE DTL	DTL o.k.	Kommt
	13:49:51.340	KW S TURBINE DTL	DTL o.k.	Schaltzustand geändert
	14:05:32.163	KW S TURBINE DTL	DTL o.k.	Schaltzustand geändert
	14:34:28.670	KW S GENERATOR REG	N.S. Lager Vibration Alarm	Schaltzustand geändert
	14:38:27.774	KW S GENERATOR REG	N.S. Lager Vibration Alarm	Alarm
	14:38:31.148	KW S GENERATOR REG	N.S. Lager Vibration Alarm	Behoben
	14:43:44.879	KW S GENERATOR REG	A.S. Lager Vibration Alarm	Schaltzustand geändert
	14:50:36.982	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:50:43.582	KW S SCHLAPPIN REC	Position Schieber 1	Offen
	14:50:47.487	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:50:52.745	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:50:57.532	KW S SCHLAPPIN REC	Position Schieber 1	Offen
	14:51:01.848	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:51:16.319	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:51:21.005	KW S SCHLAPPIN REC	Position Schieber 1	Offen
	14:51:42.126	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:51:48.275	KW S SCHLAPPIN REC	Position Schieber 1	Offen
	14:51:57.768	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:52:01.484	KW S SCHLAPPIN REC	Position Schieber 1	Offen
	14:52:04.238	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:55:22.913	KW S SCHLAPPIN REC	Position Schieber 1	Offen
	14:55:31.636	KW S SCHLAPPIN REC	Position Schieber 1	Geschlossen
	14:55:44.935	KW S SCHLAPPIN REC	Position Schieber 1	Offen

Filter: Inaktiv Modus: Nicht aktualisieren Scroll Interval: 1 Seite(n) Scroll Reihenfolge: LOG

Neue Ereignisse - zum Anzeigen Modus "Aktualisieren" wählen

Log

GOESGEN [1] - MicroSCADA

Hauptmenü Optionen Projektierung Reports Stationen Hilfe

98-01-27 11:32 380KV UW GÖSGEN Dienstag (W5)

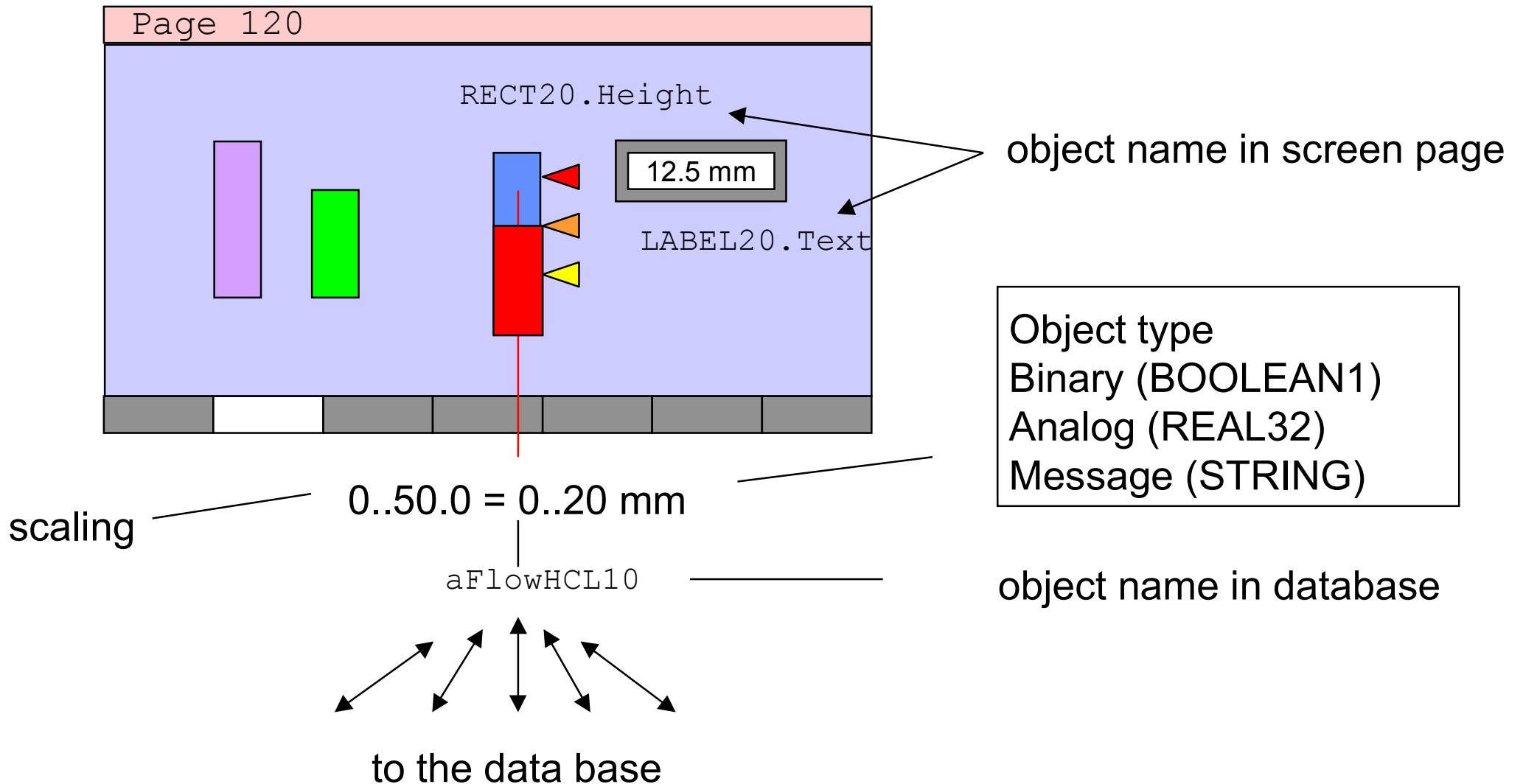
Station: 380GOES SCHALTHOHEIT
Sequenzen: Lokal WARNRUF

The diagram shows a 380KV substation with two busbars (SS1=ACS10 and SS2=ACS20) connected to seven circuit breakers (Q1-Q7). Each circuit breaker is controlled by a sequence (=ACA01 to =ACA07) and a remote control (=Fern). The busbars are connected to a transformer (=ATA31 -T31) with a 12-stage tap. The transformer is connected to a busbar (Stufe: 12) which feeds seven feeders: LAUFENBURG, KUPPELFELD, TRAFU 31, RESERVE, METTLER, KKG, and ASPHARD. The power and reactive power (P and Q) are listed for each feeder.

Feeder	P (MW)	Q (MVAR)
LAUFENBURG	2345.1	6234.1
KUPPELFELD	2345.1	6234.1
TRAFU 31	2345.1	6234.1
RESERVE	2345.1	6234.1
METTLER	2345.1	6234.1
KKG	2345.1	6234.1
ASPHARD	2345.1	6234.1

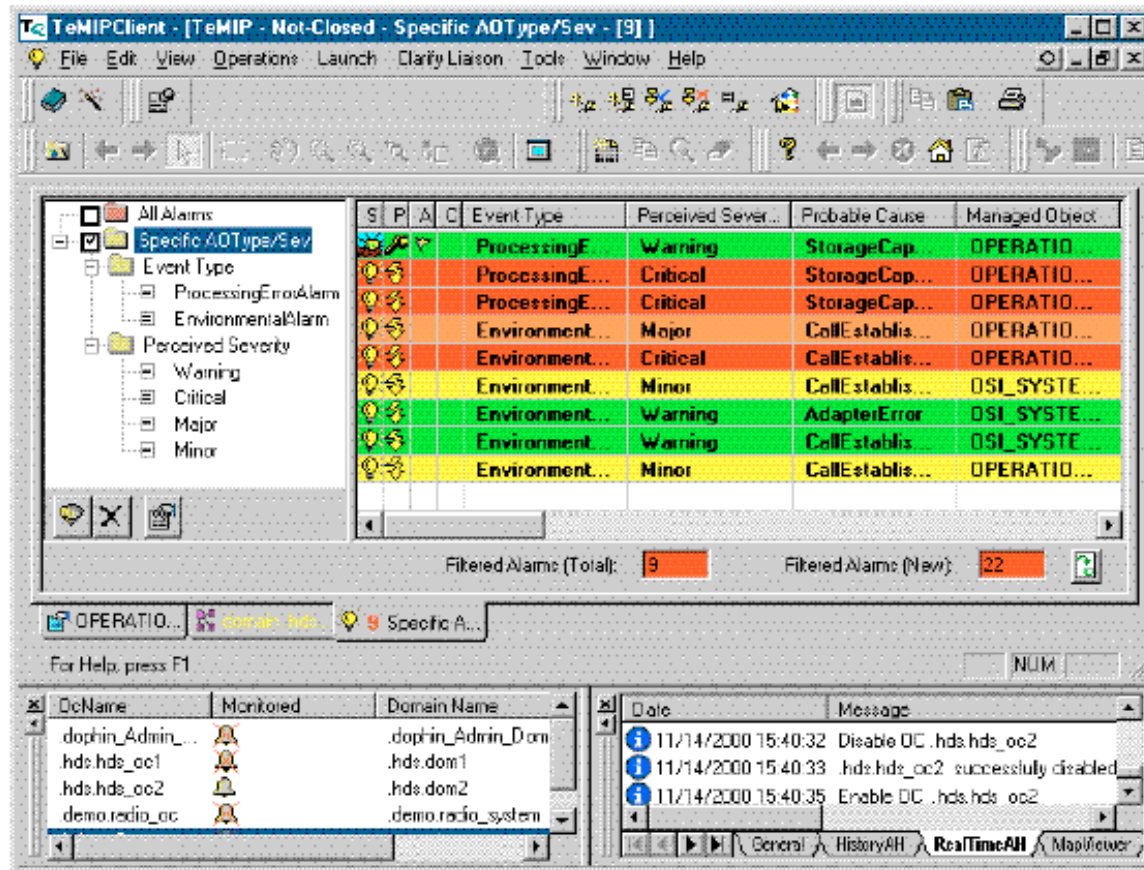
View

Binding and Scaling



Each screen object can represent several process variables....

Alarm and Event Management



time stamps exact time of arrival (or occurrence)

categorize by priorities

log for further use

acknowledge alarms

prevent multiple, same alarms

remove alarms from screen once reason disappeared (but keeps them in the log)

link to clear text explanation

What is an alarm, an event ?

A&E consider changes occurring in the plant (process) or in the control system (operator actions, configuration changes,...) that merit to be recorded.

Recorded changes can be of three kinds:

- informative: no action required
(e.g. "*production terminated at 11:09*")
- warning: plant could stop or be damaged if no corrective action is taken "soon"
(e.g. "*toner low*")
- blocking: the controller took action to protect the plant and further operation is prevented until the reason is cleared (e.g. "*paper jam*")

In general, warnings and blocking alarms should be acknowledged by the operator ("quittancer", "quittieren").

An alarm is not necessarily urgent, several levels of severity may be defined.

An event is a change related to:

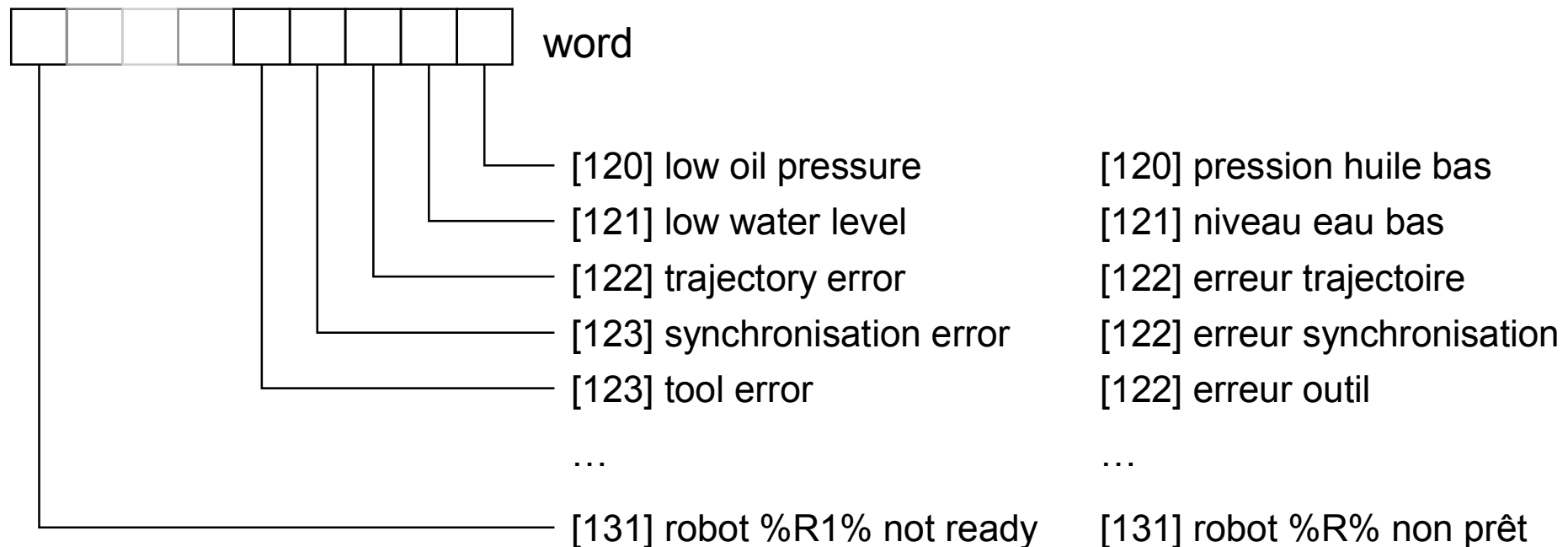
operator actions ("*grid synchronisation performed at 14:35*"),
configuration changes ("*new software loaded in controller 21*"), and
system errors ("*no life sign from controller B3*")

What triggers an alarm ?

- binary changes of process variables (individual bits),
some variables being dedicated to alarms
- reception of an analog variable that exceeds some threshold (upper limit, lower limit),
the limits being defined in the operator workstation
- reception of an alarm message (from a PLC that can generate such messages)
- computations in the operator workstation
(e.g. possible quality losses if current trend continues)
- calendar actions
(e.g. unit 233 did not get preventive maintenance for the last three months)

Implementing alarms by variables

An alarm is often encoded as a simple 16-bit word sent by an object (thru PLC) in the plant. Each bit has a different meaning, the error condition is reset when the word is 0.



This coding allows to display the error message in several national languages. A database contains the translations.

Problem: keep devices and alarm tables in the operator workstation synchronized

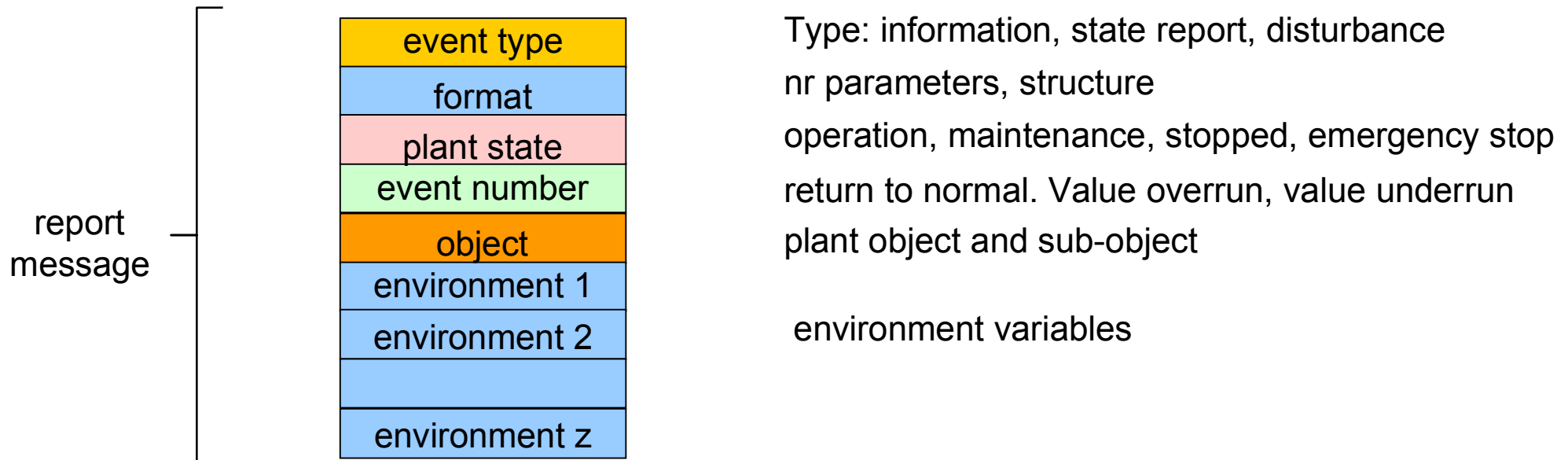
Example of a log: states, alarms,

12.3.02 13:40	Gpcpt2ofpbonne	4824	GP : Compteur 2 Ordre de Fabrication Piece bonne MD
12.3.02 13:40	Cpt2bac	50	Compteur pieces dans bac
12.3.02 13:40	Gpcpt2bac	70	
12.3.02 13:40	Gpcptbe2	45	GP Compteur pieces B equipe 2
12.3.02 13:41	Gpcpt1bac	151	
12.3.02 13:41	Gpcpt1ofpbonne	4826	GP : Compteur 1 Ordre de Fabrication Piece bonne MD
12.3.02 13:41	Gpcptae2	45	GP Compteur pieces A equipe 2
12.3.02 13:41	Cpt1bac	49	Compteur pieces dans bac
12.3.02 13:41	Gpdefr2	64	MOT32_GP
12.3.02 13:41	Gpvoydef	2	
12.3.02 13:41	Gpr3tempcycleprd	318	GP : Mot R3 Temps de Cycle de Production
12.3.02 13:42	Gpstn1e1	16	GP : [Stn1E1] Affichage des informations des présences pièces (outillage 1)
12.3.02 13:42	Gpalarme1	0	GP : Mot 1 alarme
12.3.02 13:42	Gpalarme2	0	GP : Mot 2 alarme
12.3.02 13:42	Gpstn1e1	240	GP : [Stn1E1] Affichage des informations des présences pièces (outillage 1)
12.3.02 13:43	Gpetatmodemarche	2	GP : Etat du mode de marche: MANUAL
12.3.02 13:43	Gptpscycle	1346	GP Temps de cycle cellule
12.3.02 13:43	Gpetatmodemarche	1	GP : Etat du mode de marche
12.3.02 13:43	Gpdefgene1	16	MOT1: Arret d'urgence robot 3
12.3.02 13:43	Gpetatmodemarche	0	GP : Etat du mode de marche
12.3.02 13:43	Gptpscycle	317	GP Temps de cycle cellule
12.3.02 13:43	Gpdefr2	0	MOT32_GP
12.3.02 13:43	Gpvoydef	0	
12.3.02 13:43	Gpdefgene1	0	MOT1
12.3.02 13:44	Gpetatmodemarche	1	GP : Etat du mode de marche: AUTOMATIQUE
12.3.02 13:44	Gpr2tempcycleprd	1992	GP : Mot R2 Temps de Cycle de Production
12.3.02 13:44	Gptpscycle	435	GP Temps de cycle cellule
12.3.02 13:44	Gpalarme3	1	GP : Mot 3 alarme
12.3.02 13:44	Gpalarme4	1	GP : Mot 4 alarme
12.3.02 13:44	Gpalarme3	0	GP : Mot 3 alarme
12.3.02 13:44	Gpcpt2ofpbonne	4823	GP : Compteur 2 Ordre de Fabrication Piece bonne MD

Alarm messages

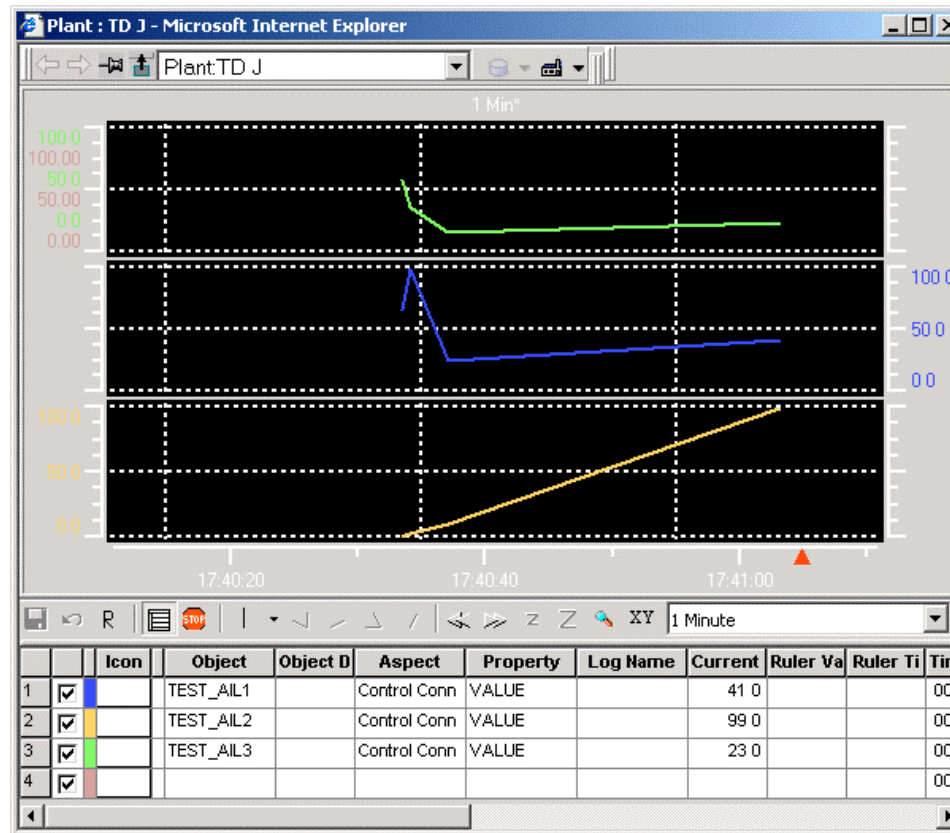
As bandwidth became available, devices can send alarm and event messages instead of alarm variables.

These messages include alarm details, and especially environment information (under which circumstances did the alarm occur)



The variable values are included when parsing the multi-lingual human-readable messages
"robot 5 on cell 31, motor 3 overheat (96°)." "robot 5 de cellule 31, moteur 3 surchauffe (96°)."

Trends



Trends allow to follow the behaviour of the plant and to monitor possible excursions. Monitored process data (sampled or event-driven) are stored in the historical database. Problem: size of the database (GB / month)

Historian

The historian keeps process relevant data at a lower granularity than the trend recorder, but with a larger quantity.

Data from different sources is aggregated in one data base, normally using data compression to keep storage costs low.

Data are analysed according to "calculation engines" to retrieve "metrics":

- performance indicators
- quality monitoring
- analysis of situations (why did batch A worked better than batch B)

Build the audit trail: "who did what, where and when"
especially in accordance with regulations (e.g. Food and Drugs Administration 's CFR 11)

Examples:

ABB's Information Manager

GE's iHistorian 2.0

Siemens's WinCC-Historian

Additional functions

printing logs and alarms (hard-copy)

reporting

display documentation and on-line help

email and SMS, voice, video (webcams)

access to databases (e.g. weather forecast)

optimisation functions

communication with other control centres

personal and production planning (can be on other workstations)

Special requirements for the food&drugs industry

The US Food&Drugs administration (FDA) requires a strict control of production for pharmaceuticals and food (FDA 21 CFR Part 11).

All process operations must be registered, the persons in charge known, the document signed (electronic signature), tamper-proof records kept.

Engineering tools

draw the objects

bind controllers to variables

define the reports and logs

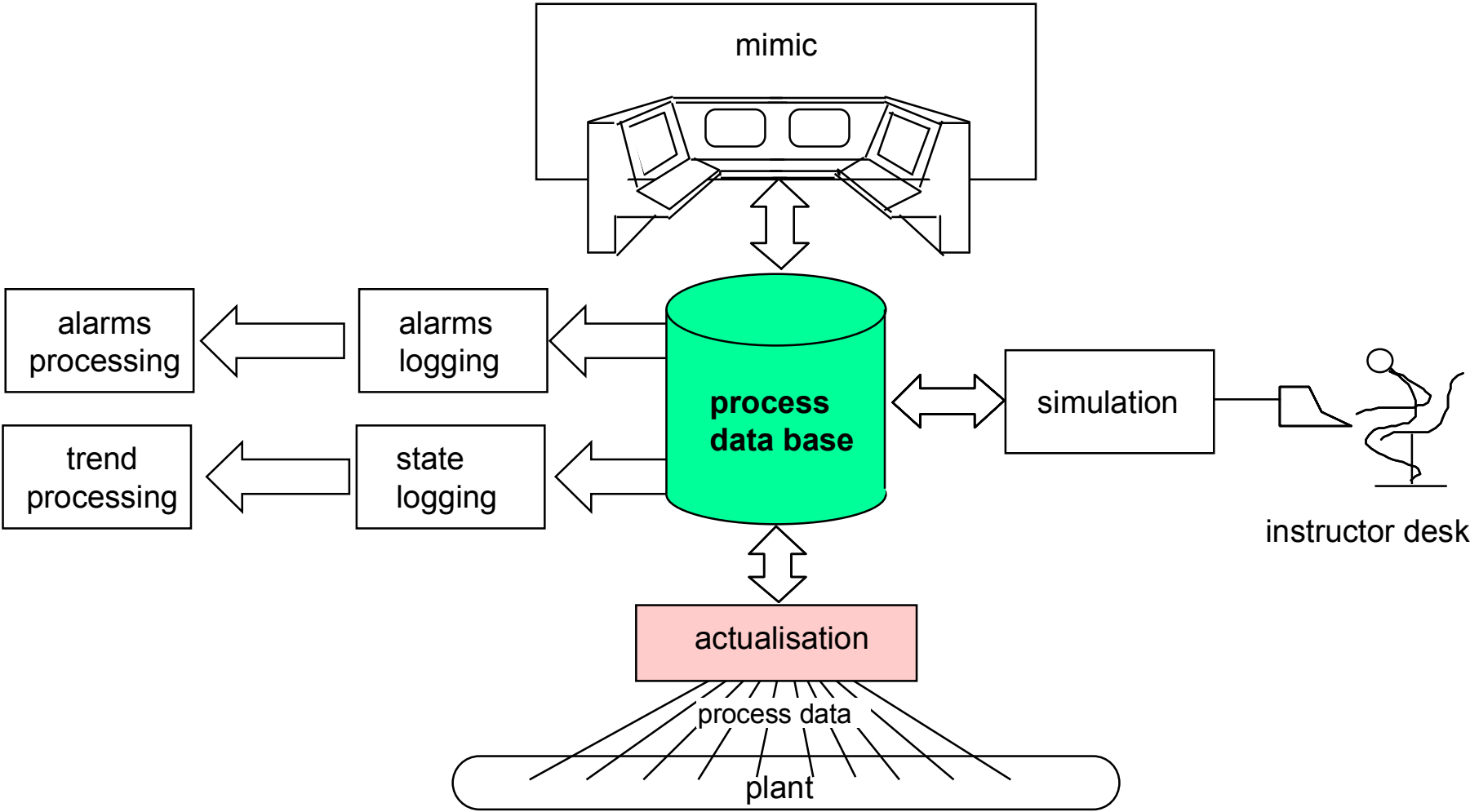
define recipes (=macros)

distribute the SCADA application (on several computers,...)

support fault-tolerance and back-ups

define interfaces to external software (SQL, SAP, etc.)

Elements of the operator workstation



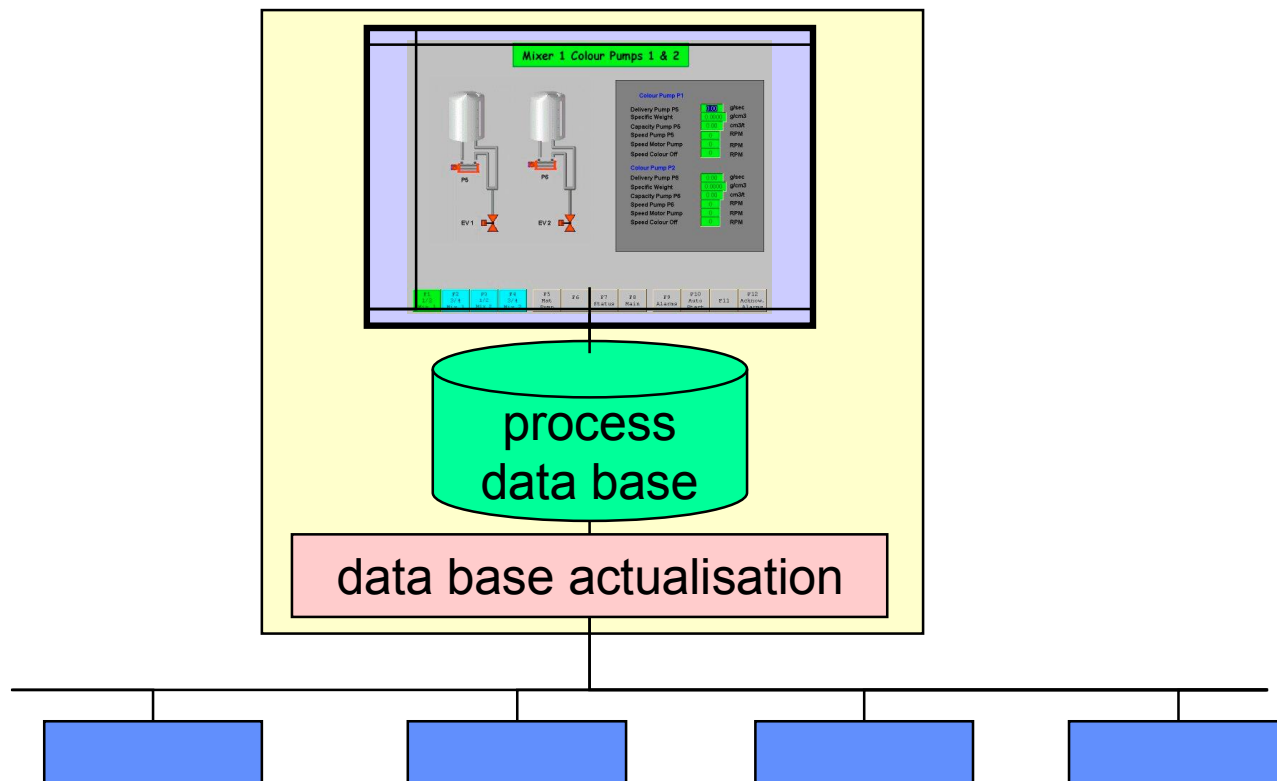
Populating the Process Data Base

Process data represent the current state of the plant.

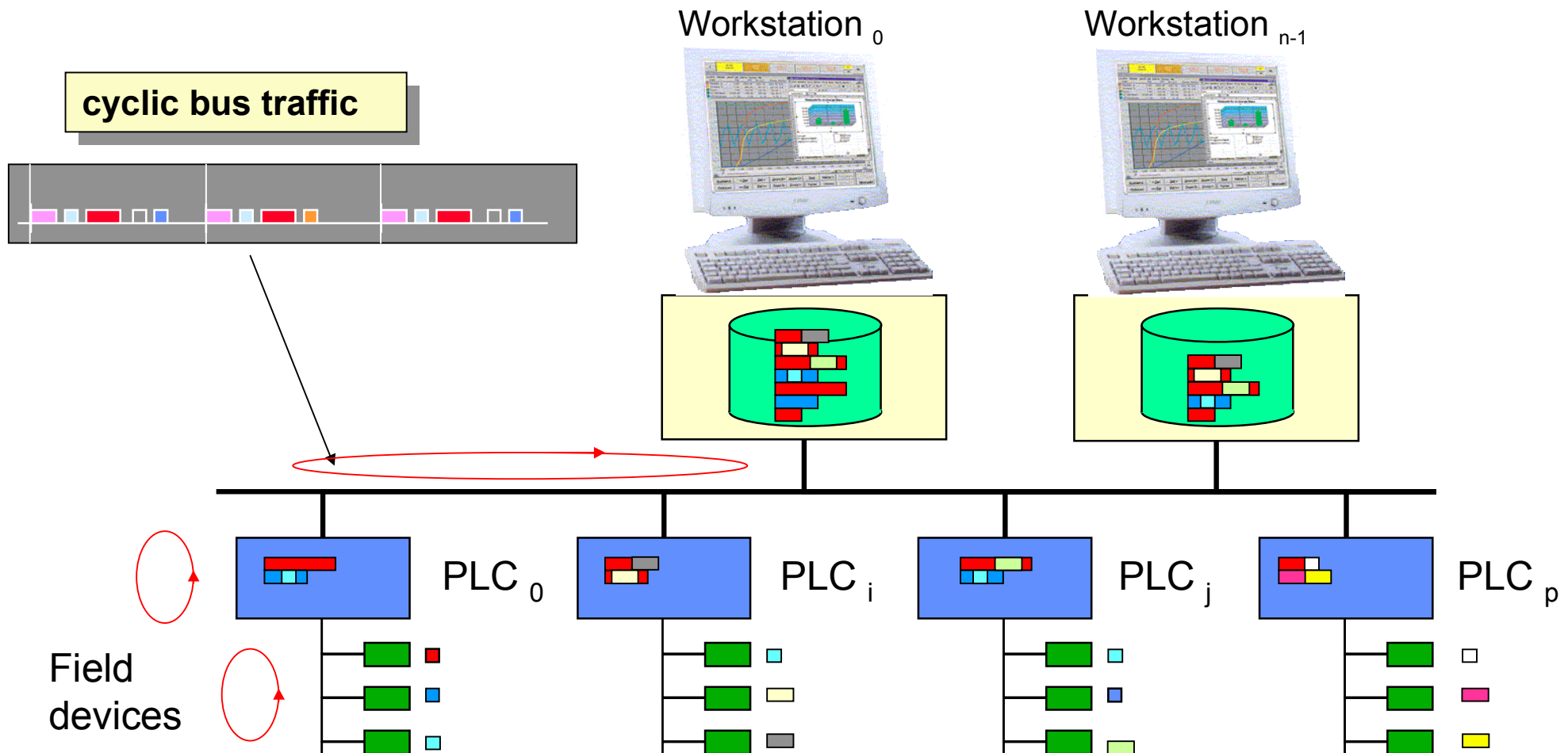
Older values are irrelevant and are overwritten by new ones ("écrasées", überschrieben)

Process data are actualized either by

- polling (the screen fetches data regularly from the database (or from the devices))
- events (the devices send data that changed to the database, which triggers the screen)



Cyclic operation

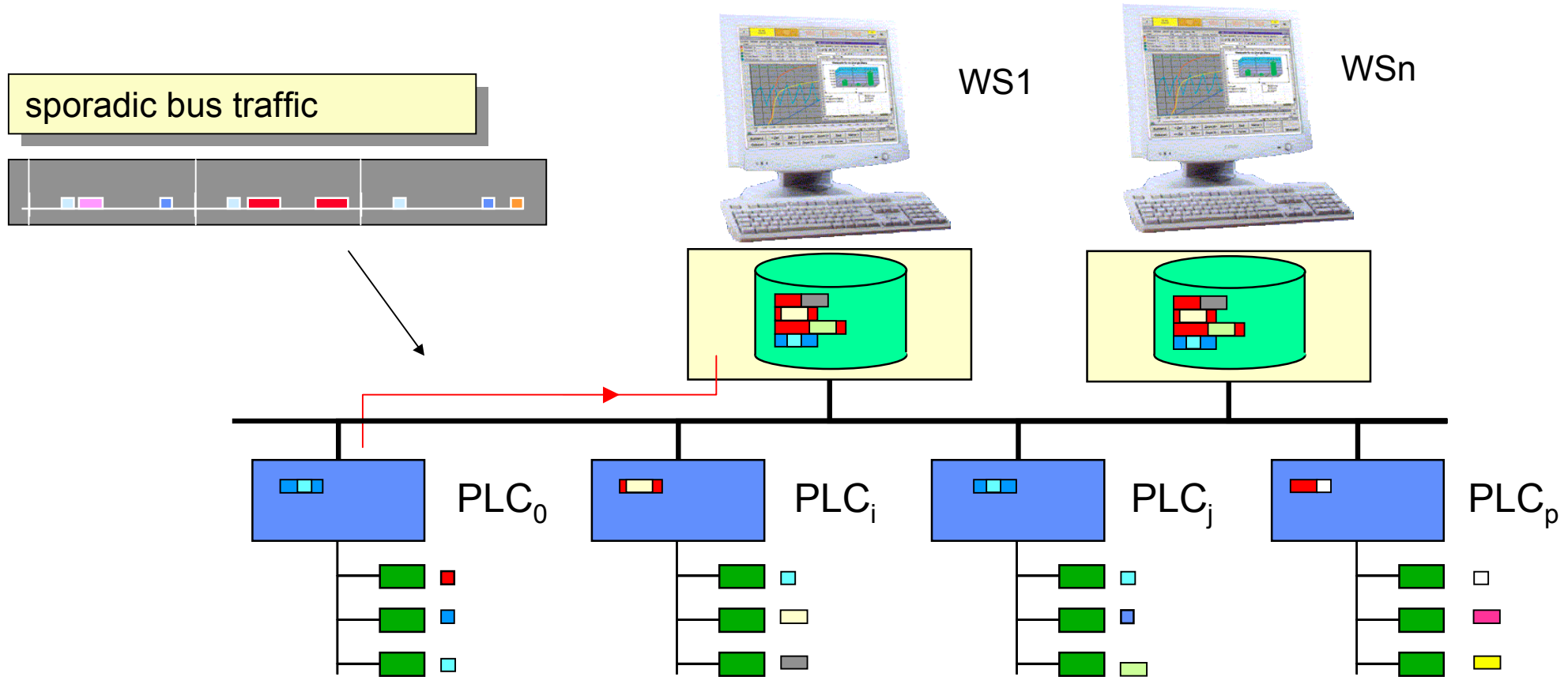


Each station broadcasts cyclically all its variables: the control bus acts as an online database
Datasets are replicated by broadcast to any number of destinations

Advantage: real-time response guaranteed

Drawback: bus bandwidth may become insufficient with large number of urgent data

Event-driven operation



Every PLC detects changes of state (events) and sends the new value over the bus

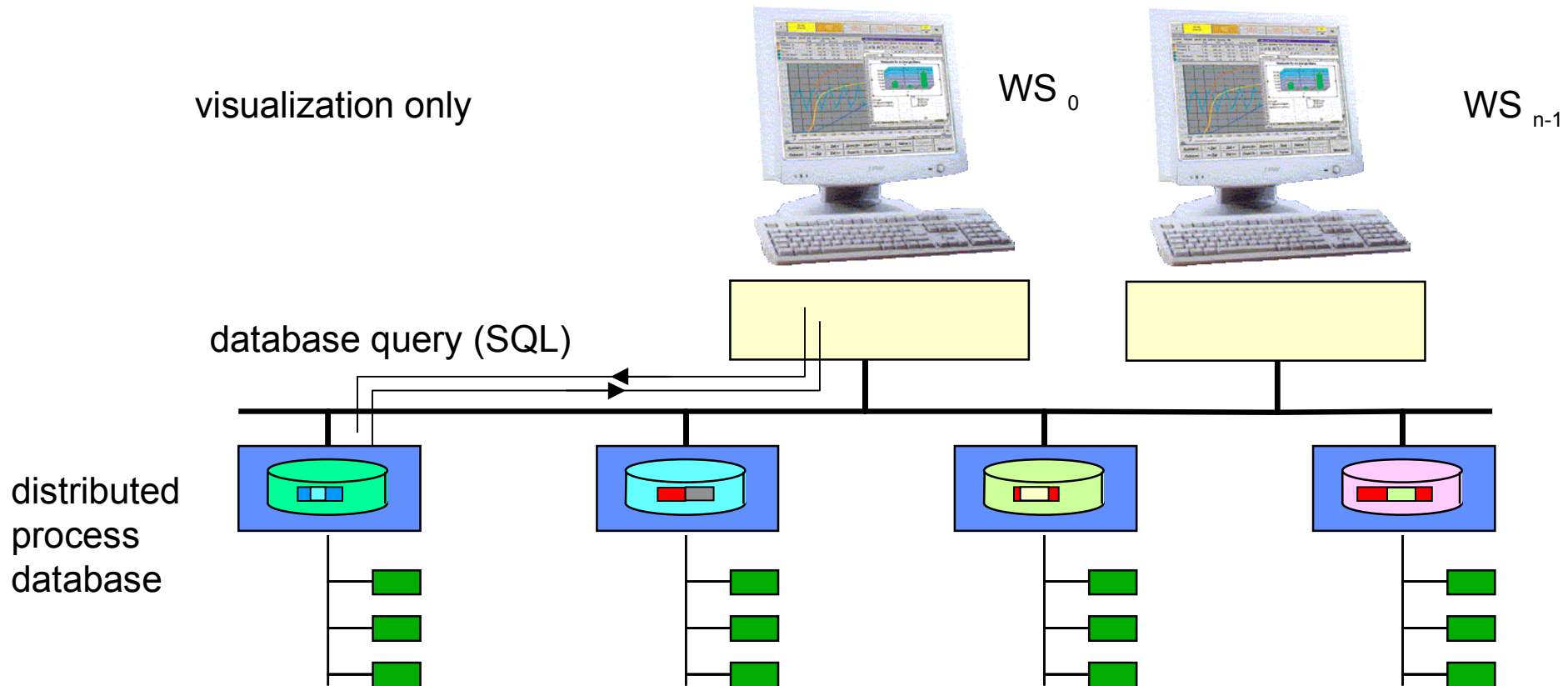
Each operator station receives and inserts data into its local database

Data are readily available for visualization

Multiple operator workstations could be addressed in multicast (acknowledged) or broadcast

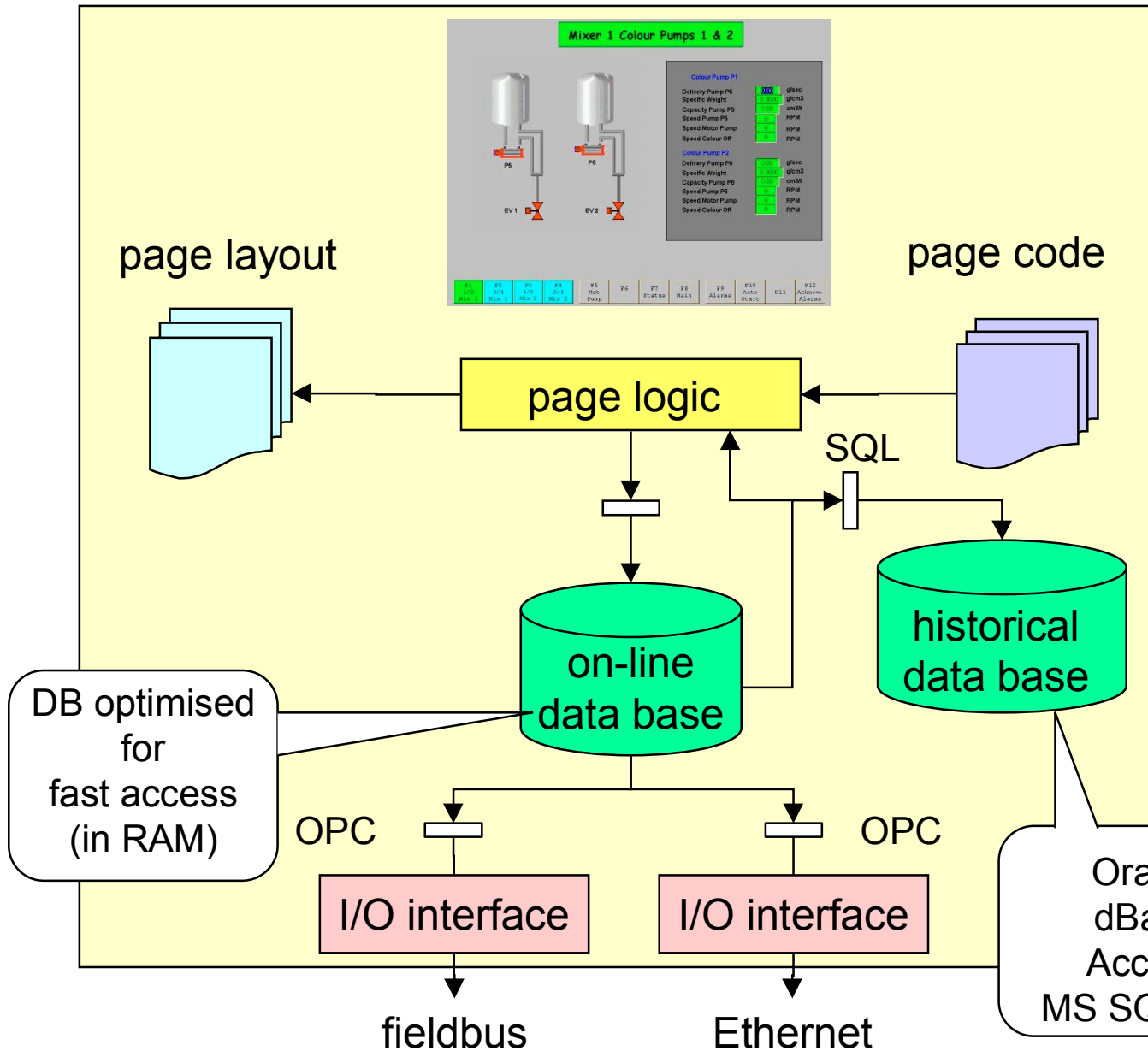
Drawback: consistency between databases, bus traffic peaks, delays

Subscription principle



To reduce bus traffic, the operator stations indicate to the controllers which data they need.
The controllers only send the required data.
The database is therefore moved to the controllers
The subscription can be replaced by a query (SQL) - this is ABB's MasterNet solution

Operator Workstation design



Graphical User Interface
access by Keyboard, Mouse,
 Trackball, Touch screen, Light
 pen

display of values, colours, shape
 depending on variable value

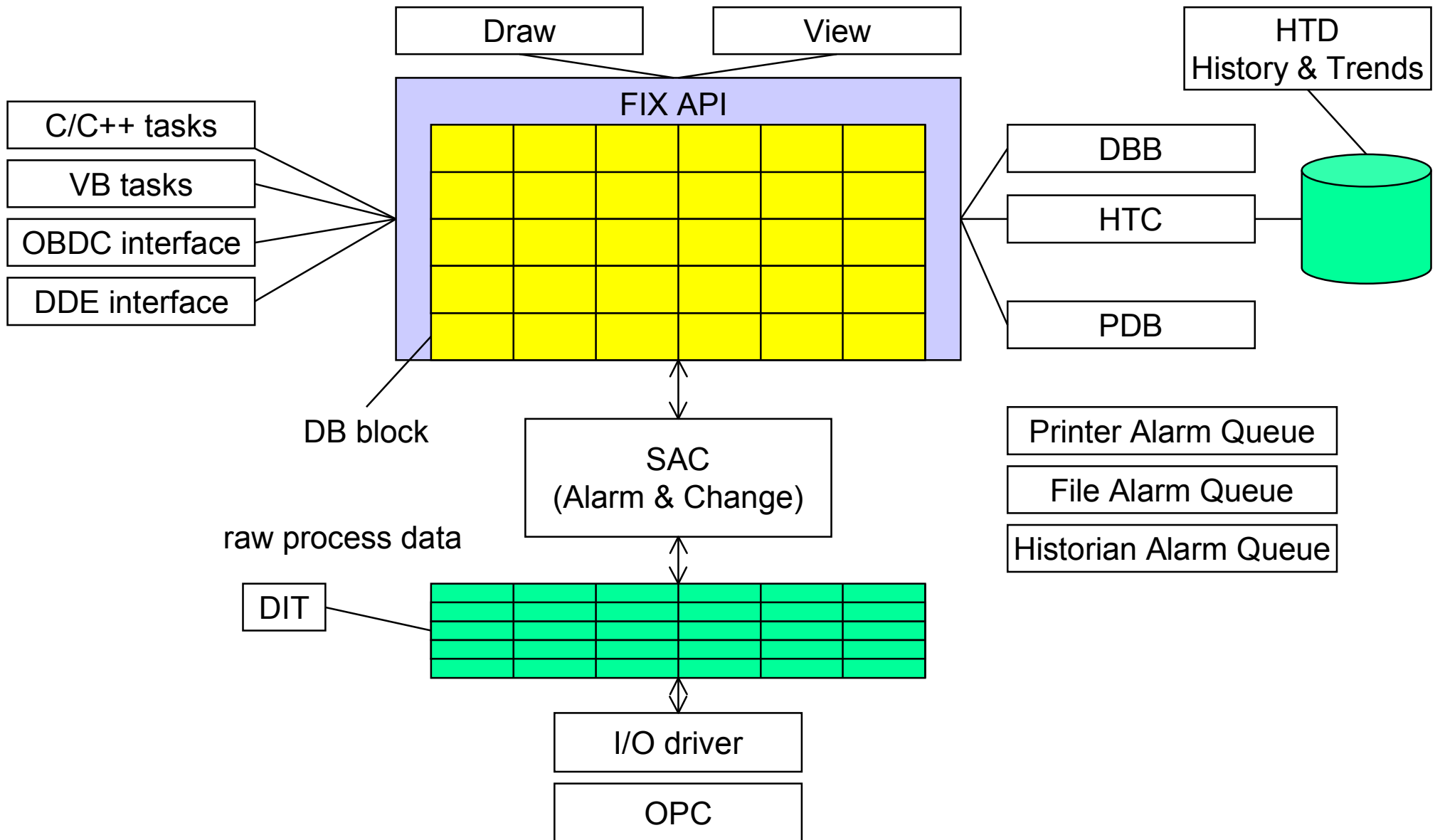
operations on visual objects
 (scaling, combination, events)
 and on acting objects
 (page change, sequence of
 events,..)

navigation from page to page
 (hierarchical, shortcuts, search,..)

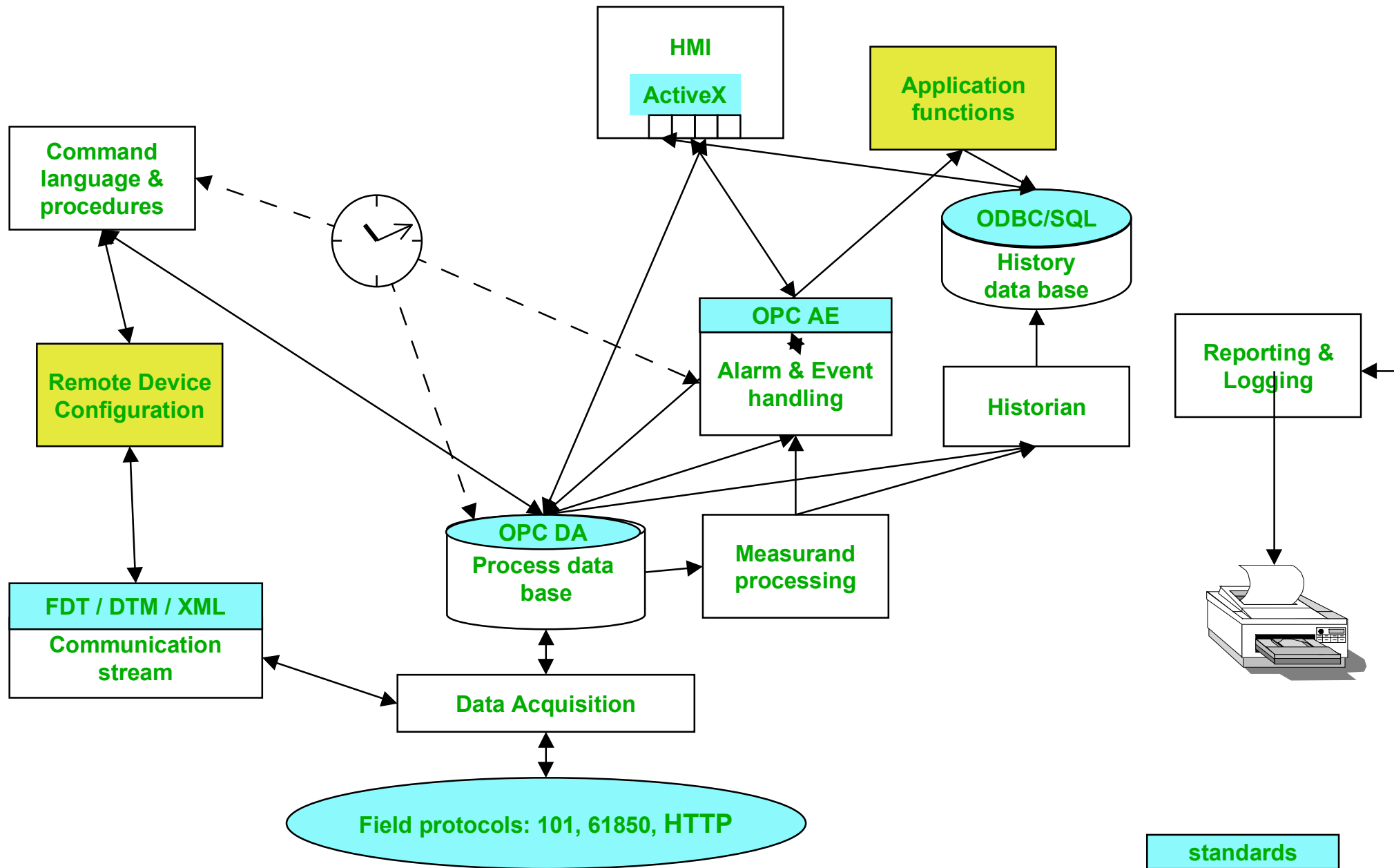
DB optimised
 for
 fast access
 (in RAM)

Oracle
 dBase
 Access
 MS SQL,

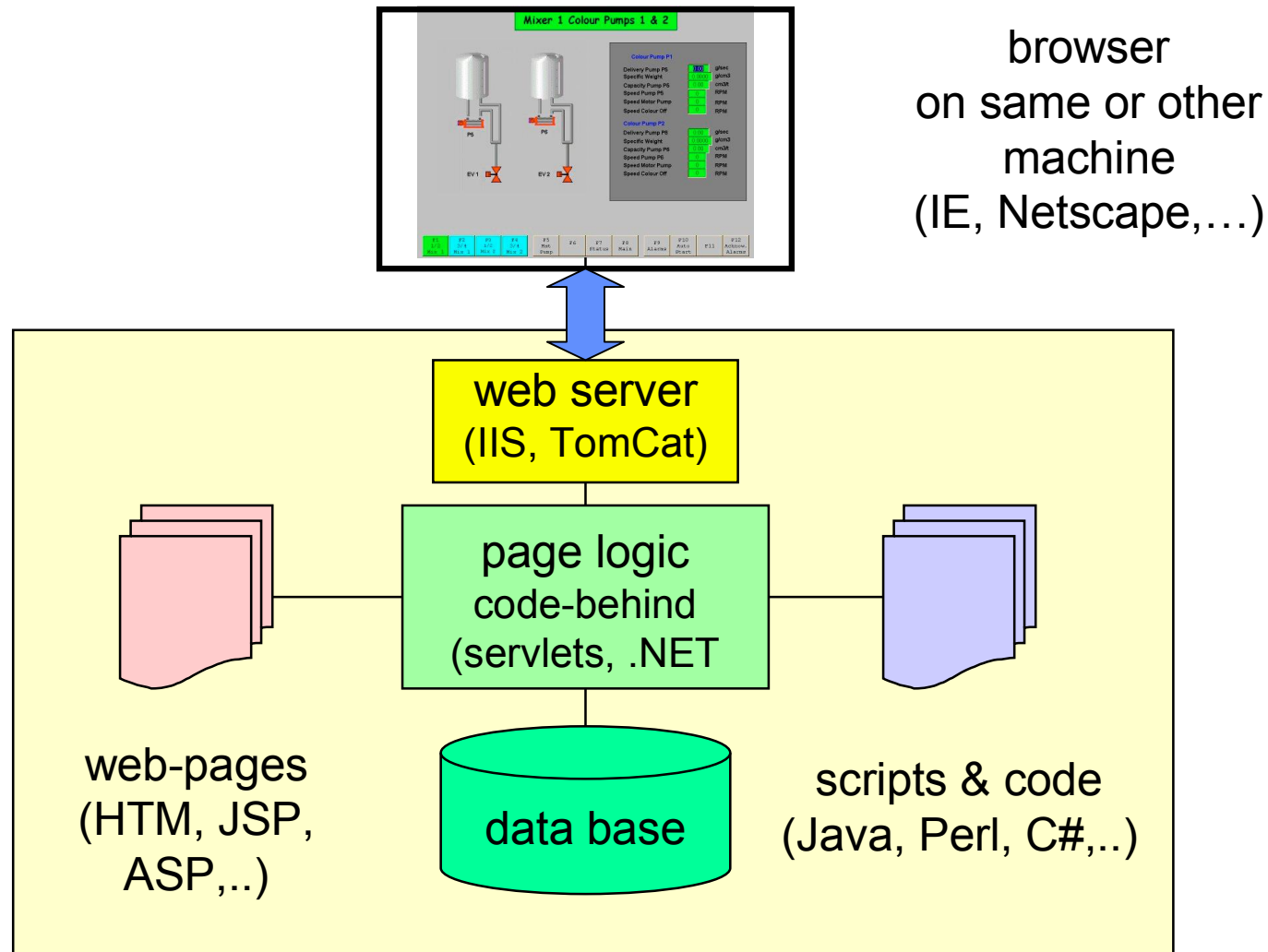
Example: Intellution's Fix32 internal structure



Scada SW architecture



Model-Viewer-Controller: from E-commerce to Industry Operator Screen



browser
on same or other
machine
(IE, Netscape,...)

the basic structure is the same....

...and why not simply Microsoft .NET ?

The value of the visualization tools is not in the basic platform (which is often Microsoft, Java, .NET or similar) ...

... it is in the conglomerate of tools and interface to different control systems they offer.

Some (Iconics) offer a library of ActiveX - Controls representing automation objects.

Protocols to a number of commercial PLCs are needed (ABB, Siemens, GE,...)

There is a growing similarity between products for SCADA and for E-commerce, but each is optimised for another market.

Why not Enterprise platform ?

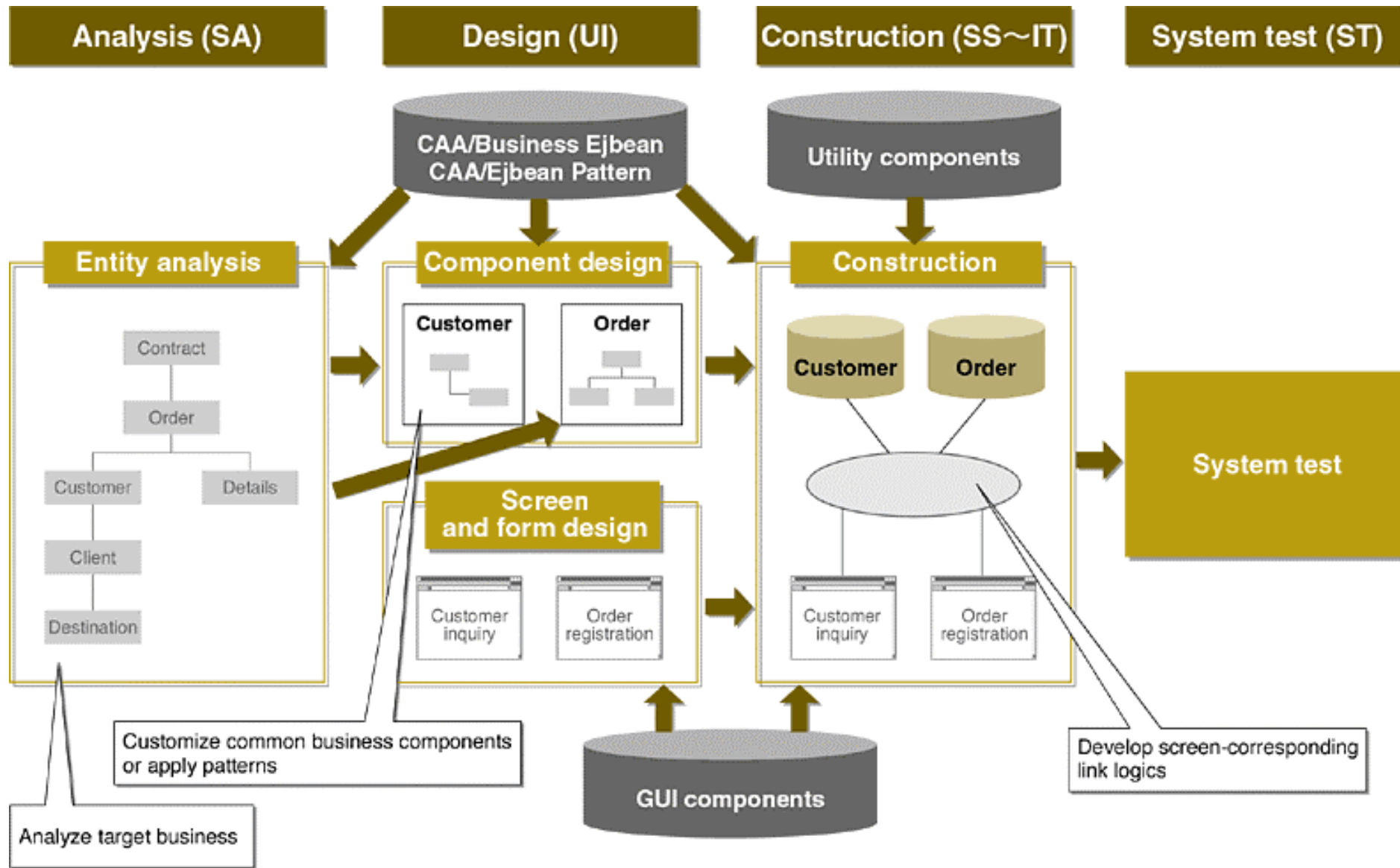


Figure 2. Confirming, selecting, and applying component units (Class chart by UML description).

An example of SCADA requirements

Action is based on production batches, signing in a new batch, identifying the paper material, filling good and responsible machine driver.

Connection to Mitsubishi A series and Siemens S7 PLCs, with asynchronous or Ethernet cable.

Connection to asynchronous ASCII-protocol communication devices for example F&P Bailey FillMag.

Process diagrams 4-5 pcs. including dynamic displays for valves and cylinders 40-50 pcs.,
motors 20 pcs., heaters 20 pcs., thermocouple-inputs 30-40 pcs.,

additional analog inputs 10 pcs.

Real time and historical trends 40-50 pcs.

Sequence displays including step-displays and clocks.

Alarm displays with additional help displays including text and pictures.

Parameter set displays for PID-controls, filling automates and servo drives.

Storing logged data to a transferable database.

quite different from E-commerce, but the platform could be the same...

Generic visualization packages

Company	Product
ABB	Process Portal, OperatorIT
CTC Parker Automation	interact
Citect	CitectSCADA (AUS, ex CI technologies, www.citect.com)
Intellution (GE Fanuc)	Intellution (iFix3.0) 65000 installs, M\$38 turnover
Iconics	Genesis
National Instruments	LabView, Lookout
Rockwell Software	RSView
Siemens	WinCC, ProTool/Pro
Taylor	Process Windows
TCP	SmartScreen
USDATA	Factorylink, 25000 installs, M\$28 turnover
Wonderware (Invensys)	InTouch , 48000 installs, M\$55 turnover

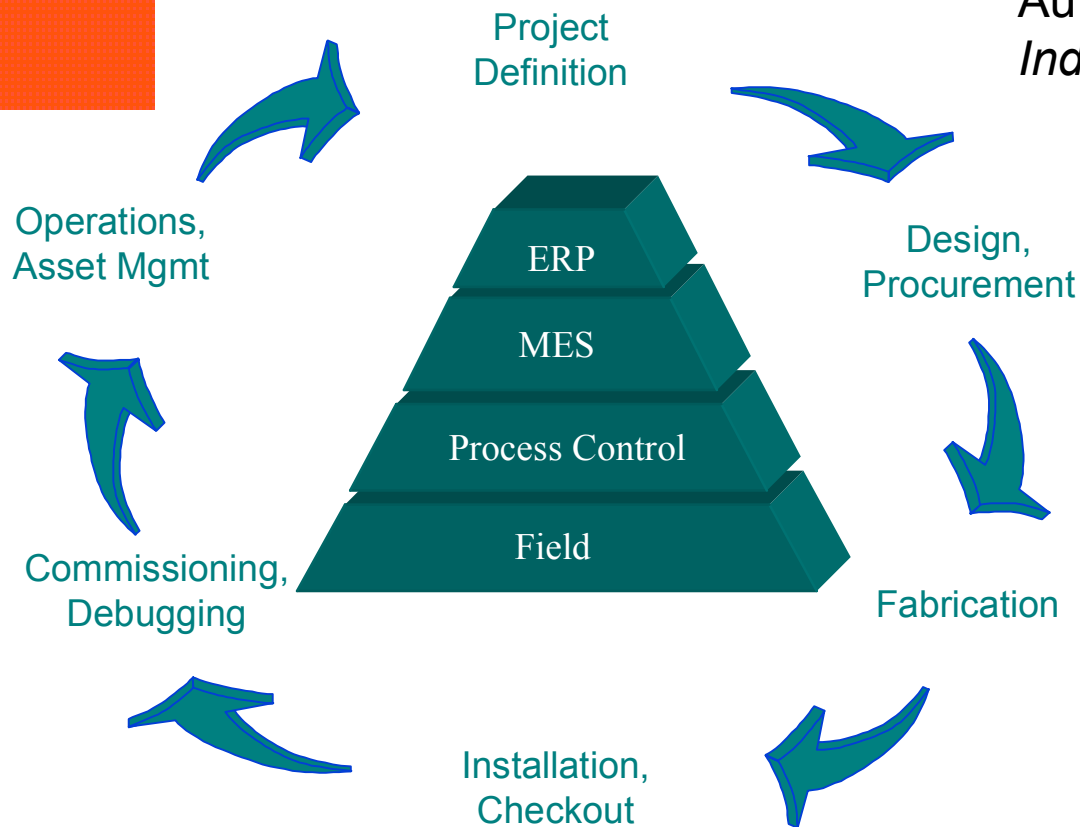
...XYCOM, Nematron, [Modicon PanelMate](#), [OIL System PI Data Historian](#).

Ann Arbor Technology, Axeda, Eaton Cutler-Hammer, ei3, InduSoft, Opto22,





Industrial Automation
Automation Industrielle
Industrielle Automation



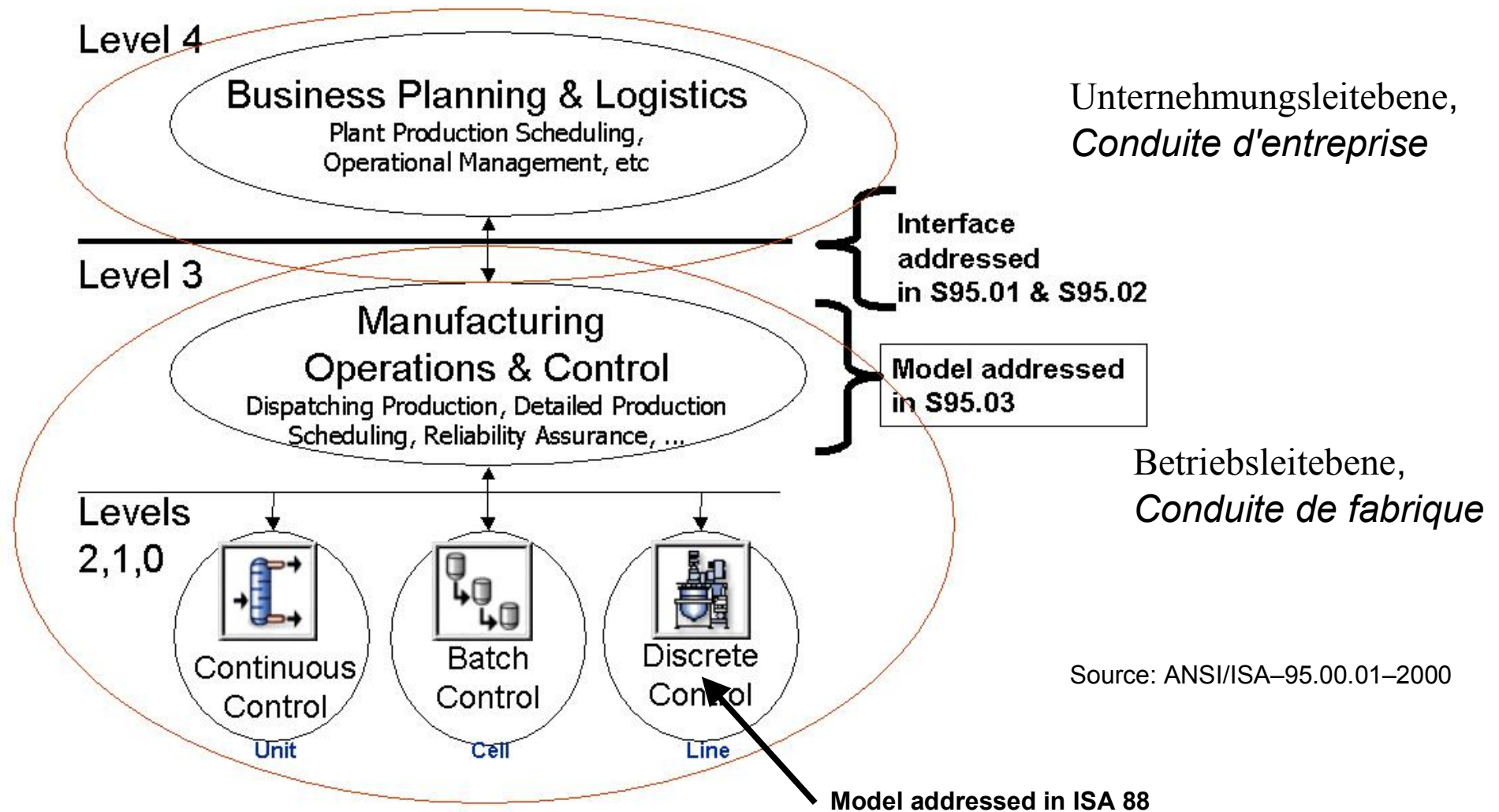
6.2

Manufacturing Execution System = MES
Pilotage de fabrication
Herstellungstechnik

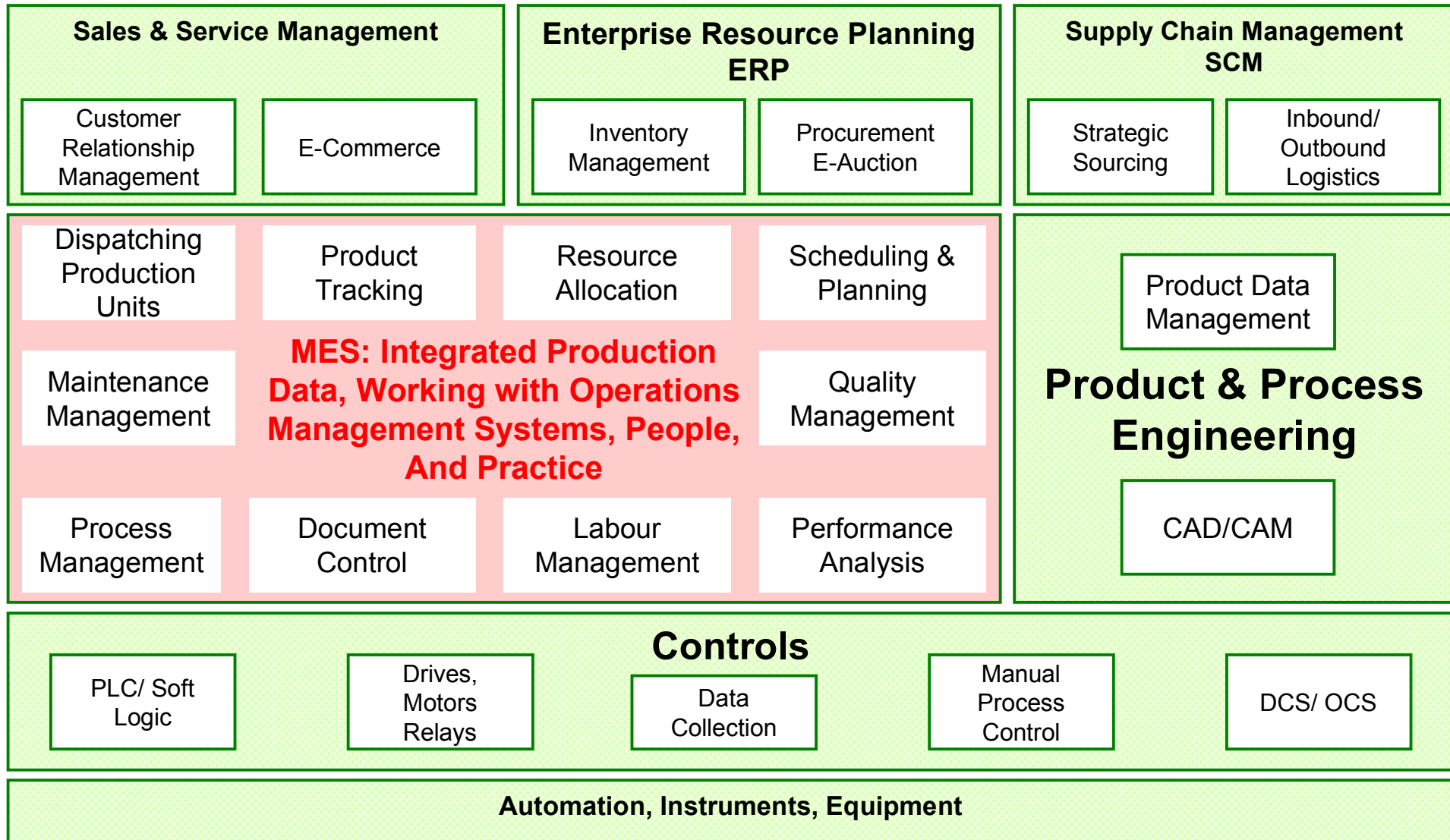
Prof. Dr. H. Kirrmann
ABB Research Center, Baden, Switzerland

Manufacturing Execution System

MES is the intermediate layer (3) between Control (0,1,2) and Enterprise (4)

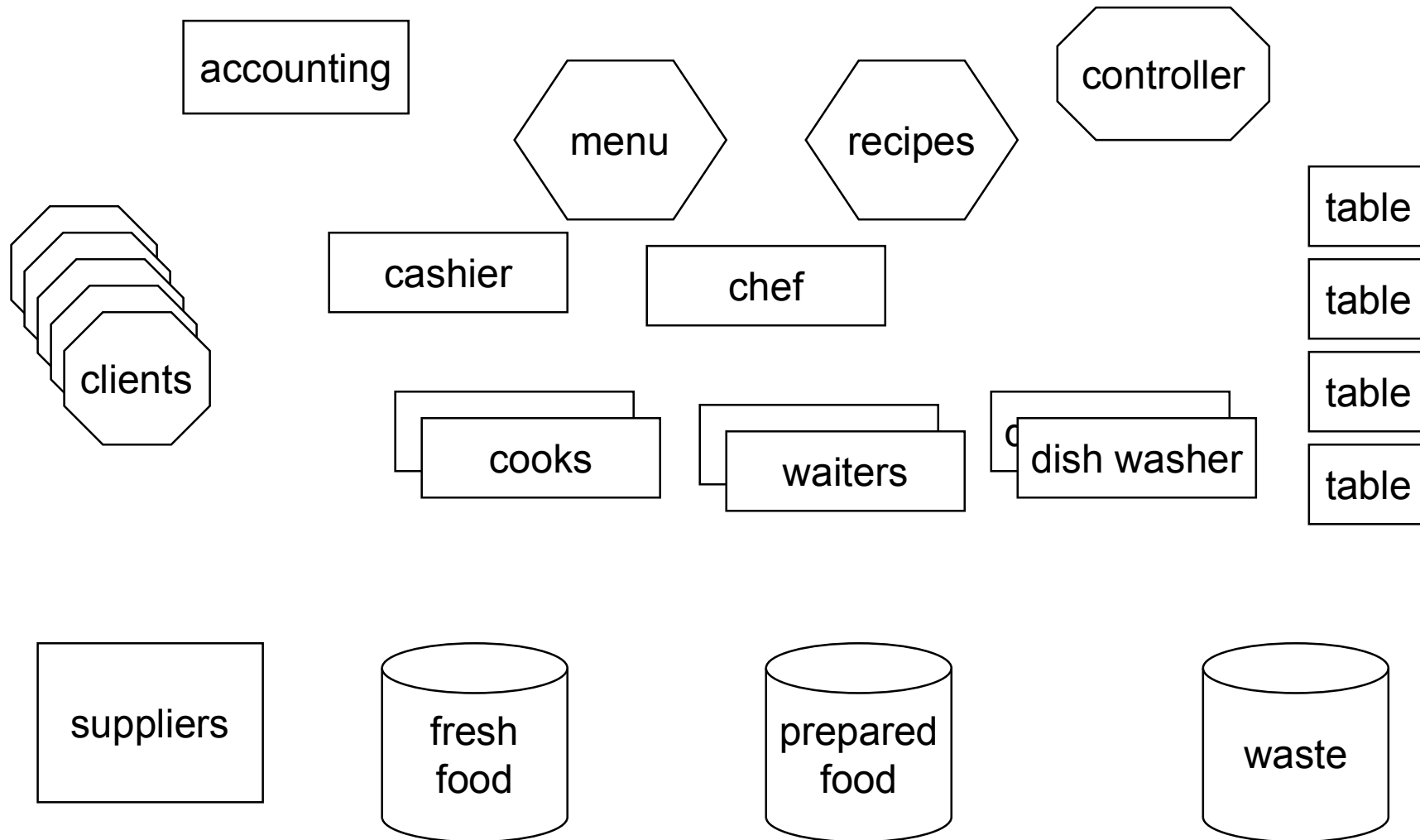


Location of MES in the control hierarchy



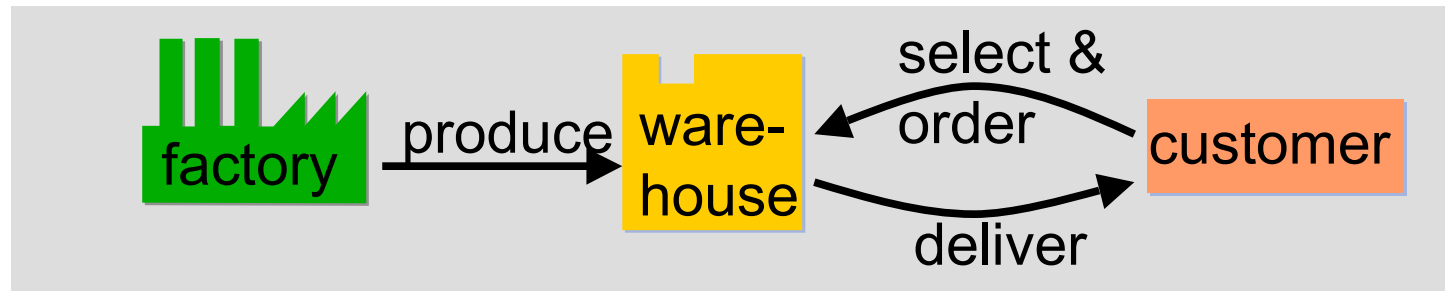
source: MESA White Paper

Manufacturing model: Restaurant

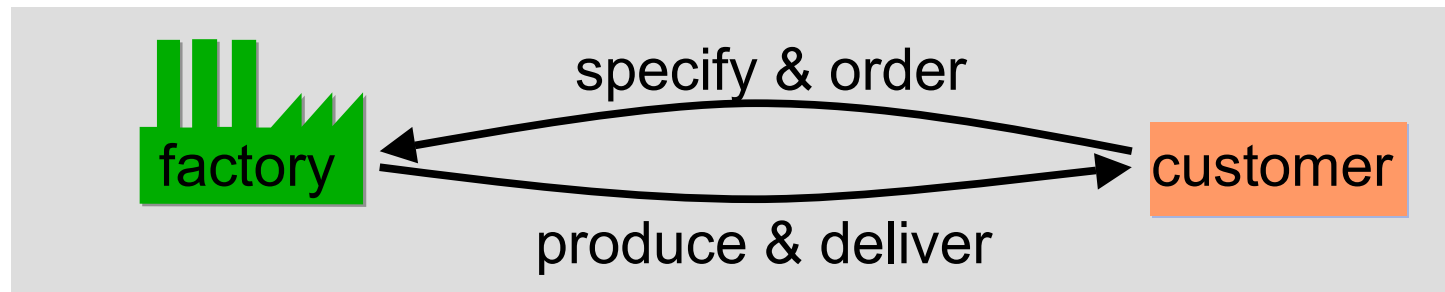


Type of production

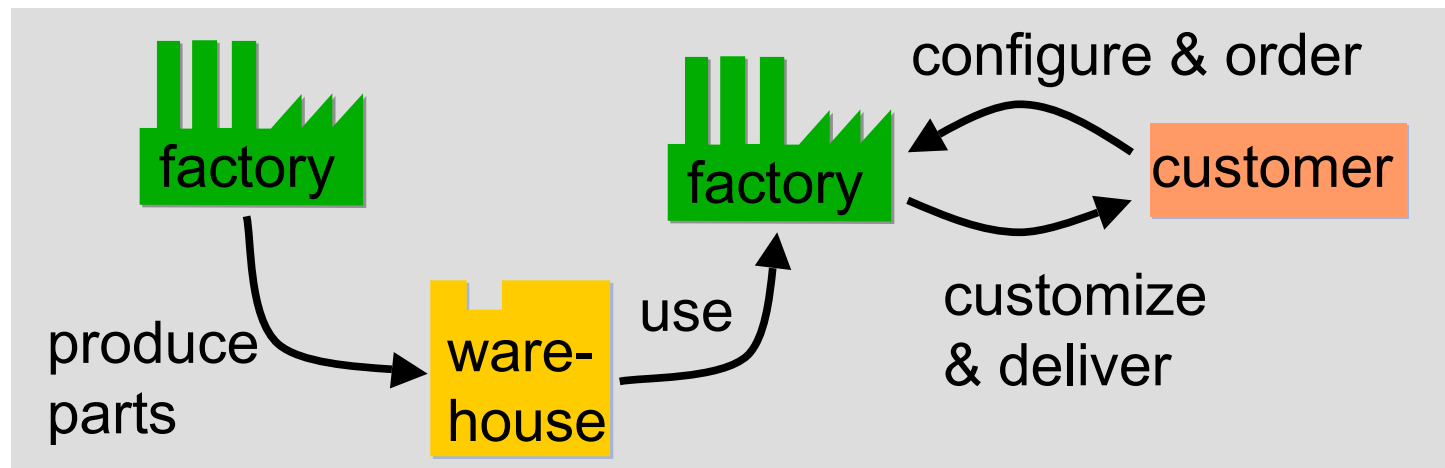
make to stock



make to order



make to configuration



Notions

Serial number: a unique identifier assigned to a produced good, lot or part

Bill of Material (BOM): list of parts and consumables needed to produce a product

Recipe: the operations needs to produce a part

Bill of Resources; non consumable resources required for production

Workflow: the flow of parts within the factory

Traceability: ability to track where the parts a product come from and who assembled them

Work Order: order to produce a certain quantity of a product

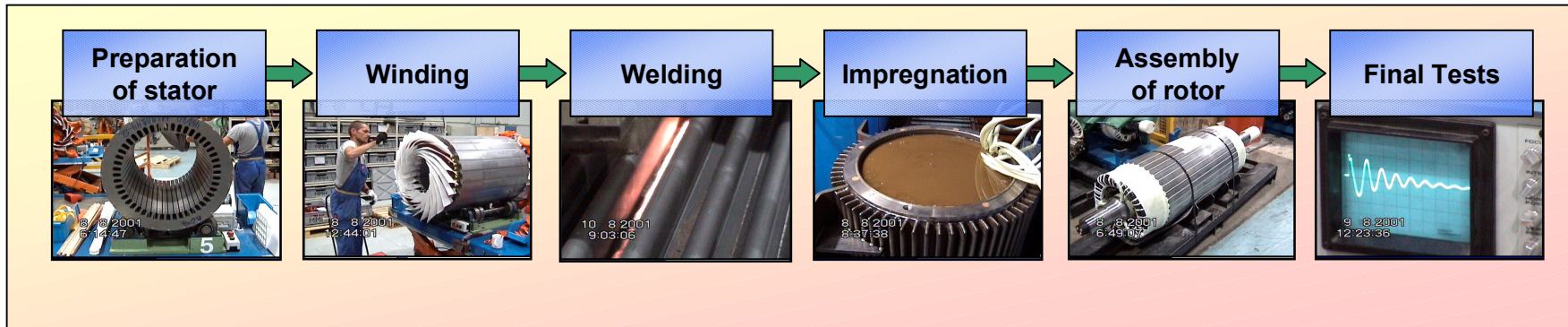
Push / pull: produce when parts are available, require parts when product is required

Kanban: supplier cares that the parts tray of the client are never void.

Scheduling / dispatching: (flight timetable / tower) (Planer / Disponent)

Engineering Change Order (ECO): design or recipe errors reported to engineering.

Example Workflow

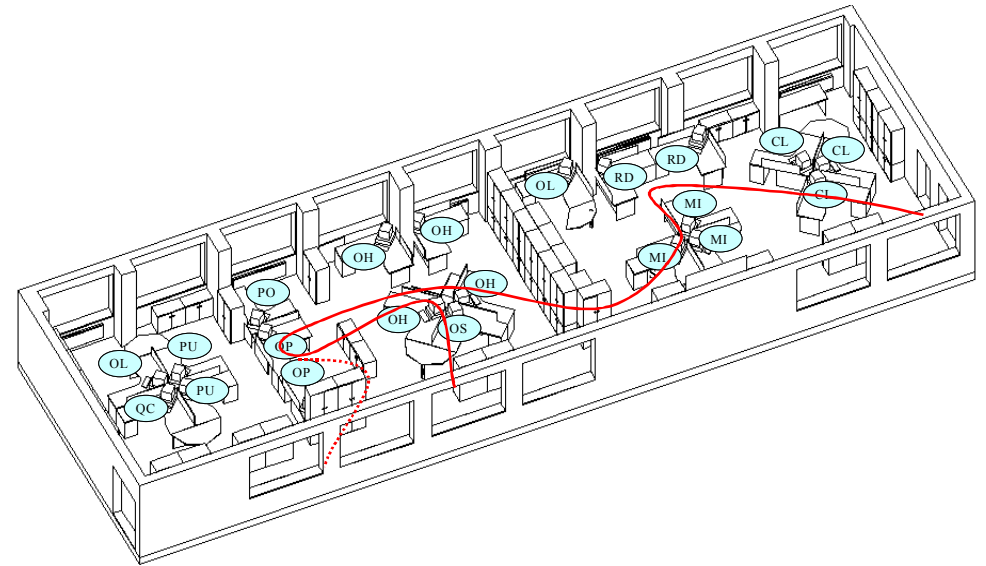
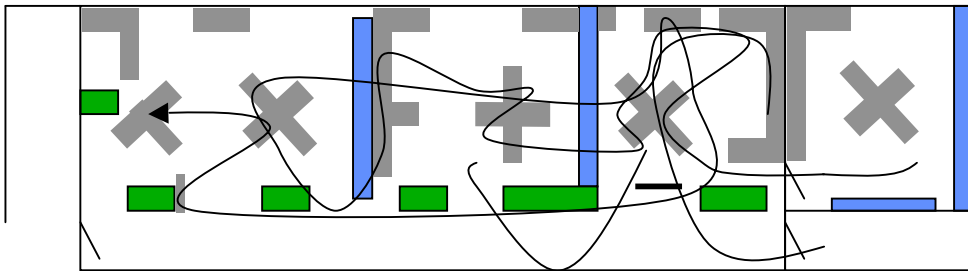


Workflow is the path that the product being manufactured takes through several “stations”.

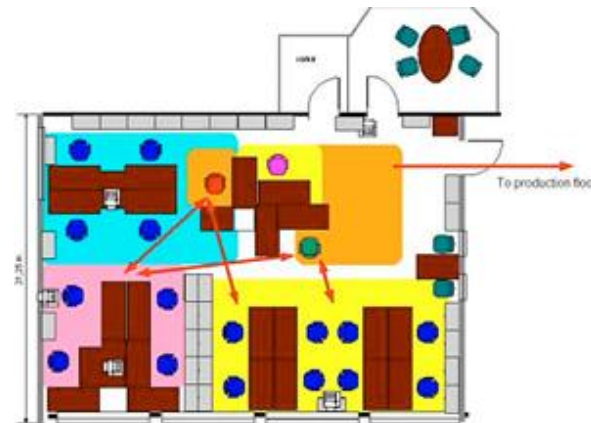
Recipe is the sequence of operations that takes place at one particular station.

Office lay-outs impact order lead-time

	Before	Current	Future
People	11	6	5
Distance	110 m	30 m	20 m
Time	70 hours	23 hours	7,5 hours



Before: ~ 300 meters (3 floors)
Time: + 4 days



After: 9 meters
Time: 2 hours

ISA S95 standard

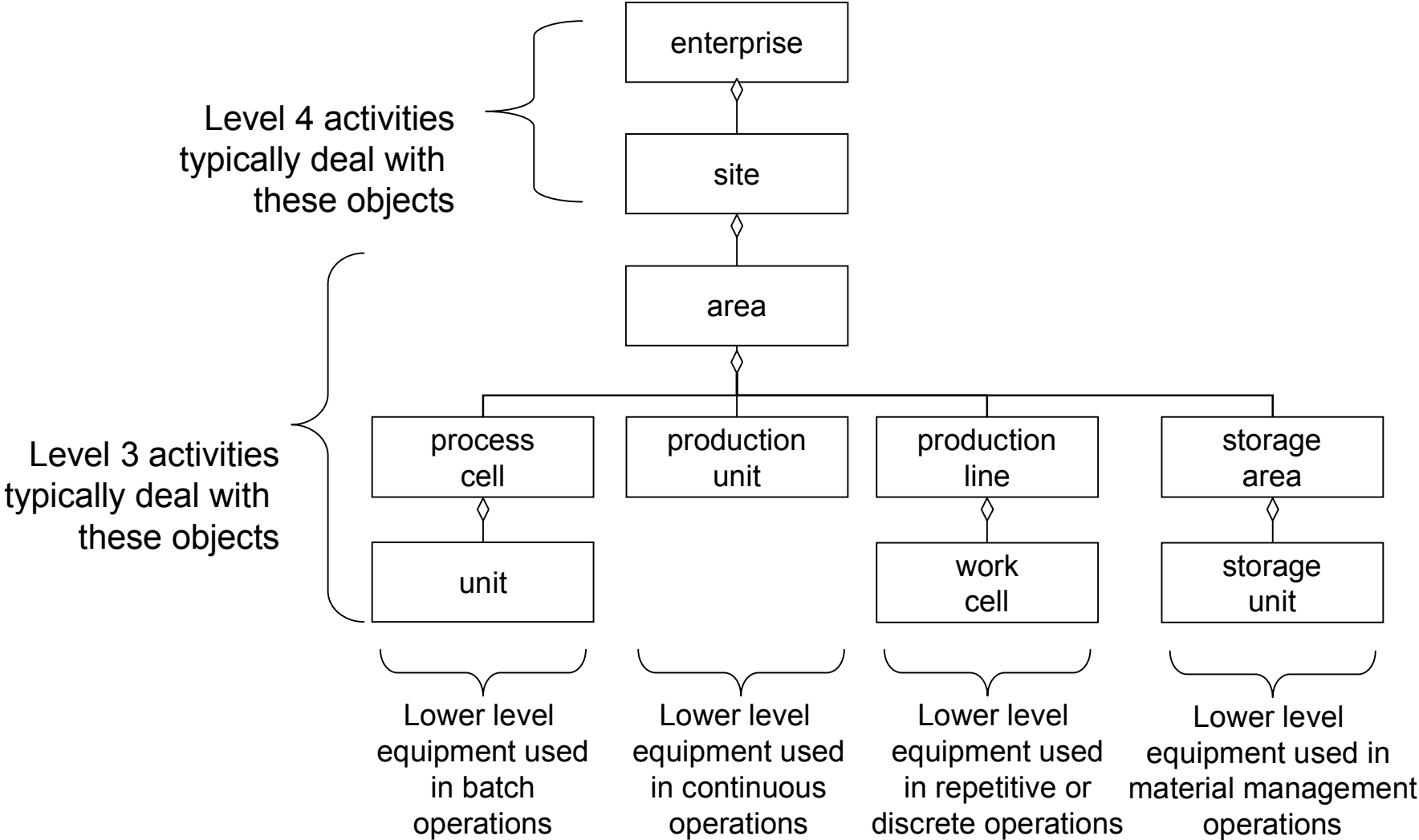
This US standard defines terminology and good practices

- Delineate the business processes from the manufacturing processes
- Identify the responsibilities and functions in Business to Manufacturing and Manufacturing to Manufacturing integration
- Identify exchanged information in Business to Manufacturing integration
- Improve integration of manufacturing by defining:
 - Common terminology
 - Consistent set of models
- Establish common points for the integration of manufacturing systems with other enterprise systems

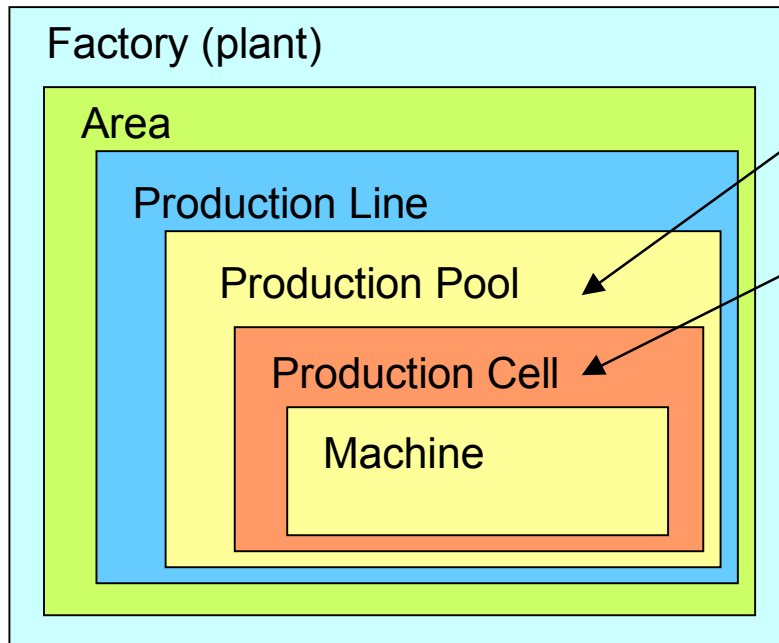
ANSI/ISA 95 standard documents

- ANSI/ISA95.00.01 “Enterprise - Control System Integration - Part 1: Models and Terminology”
 - Approved July 2000
 - IEC/ISO 62264-1 international standard approved and released by IEC/ISO
- ANSI/ISA95.00.02 “Enterprise - Control System Integration - Part 2: Data Structures and Attributes”
 - Approved October 2001
 - IEC/ISO 62264-2 international standard currently being reviewed by Joint Working Group
- Draft standards dS95.00.03 “Enterprise - Control System Integration - Part 3: Models of Manufacturing Operations”
 - Still under construction – Draft 14 released for review

Location hierarchy



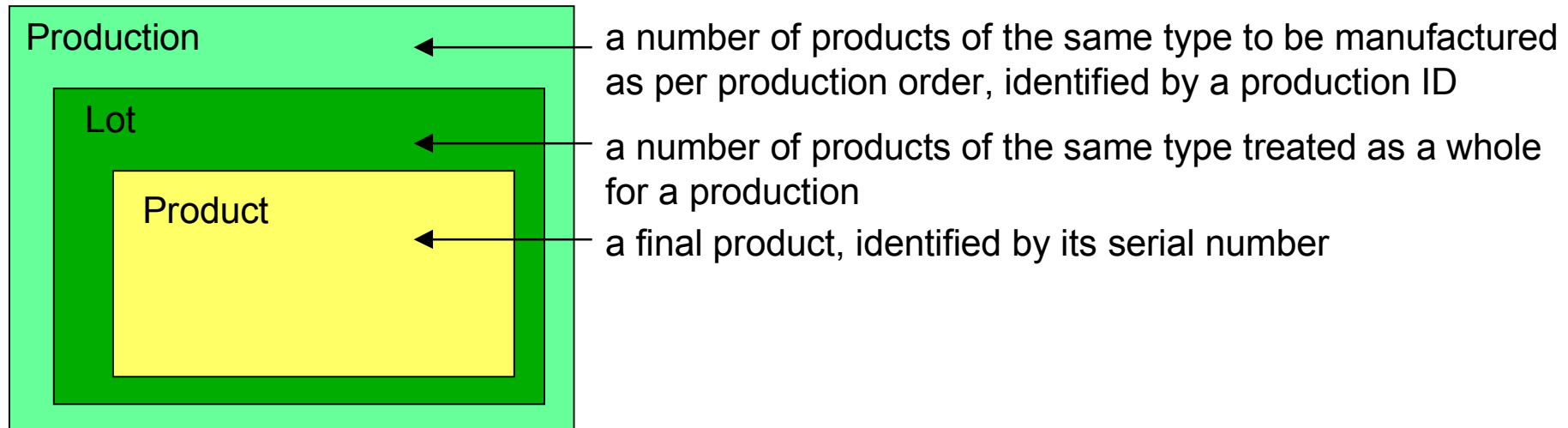
Location elements



group of production cells with identical production capabilities

a place where a particular manufacturing operation on the product is executed.

Production elements



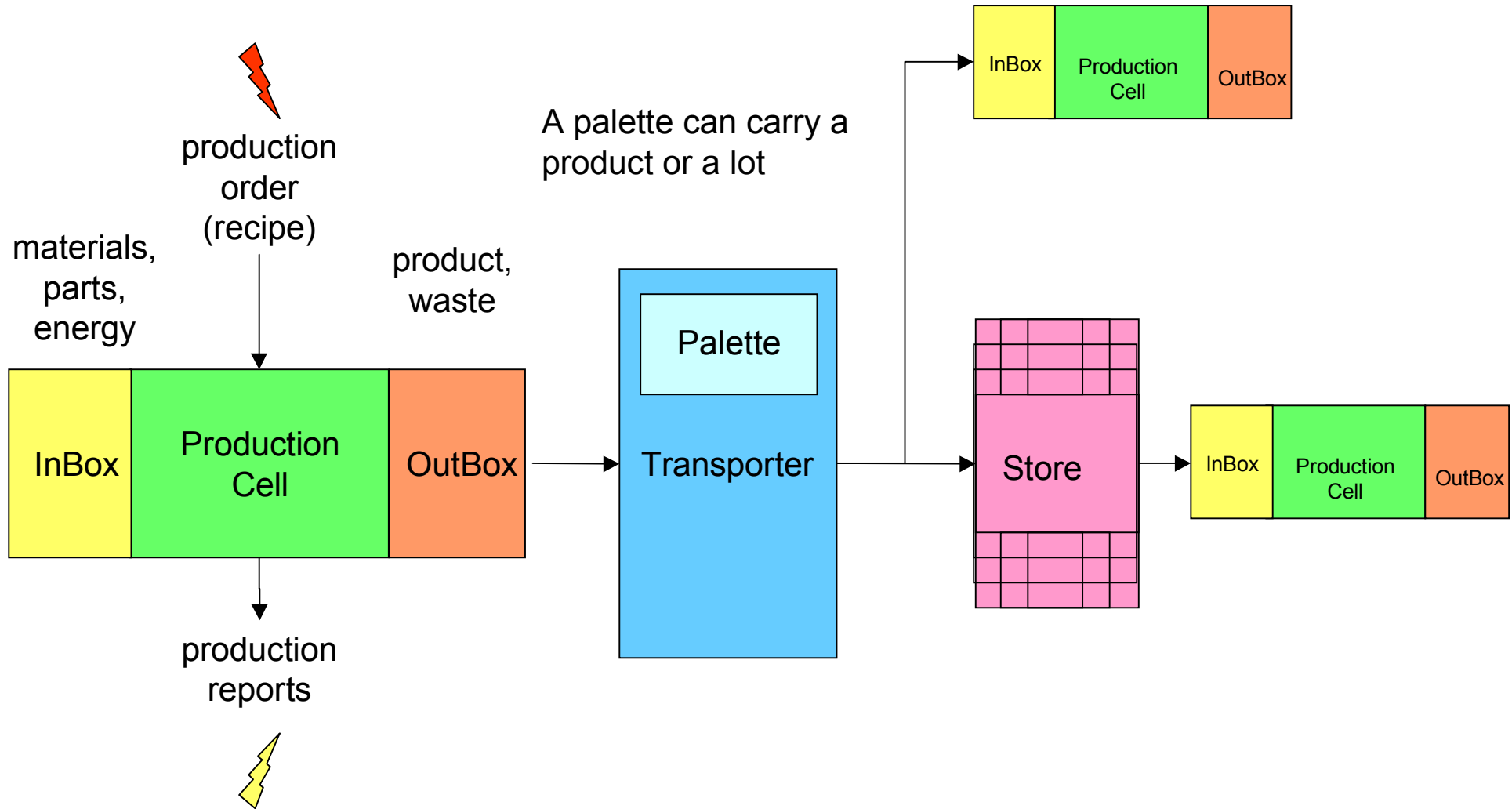
Part

identifiable components of the product,
can be used for product tracking

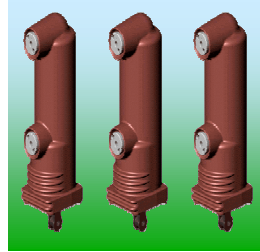
Material

expendible, not individualized components of the product

Manufacturing Elements

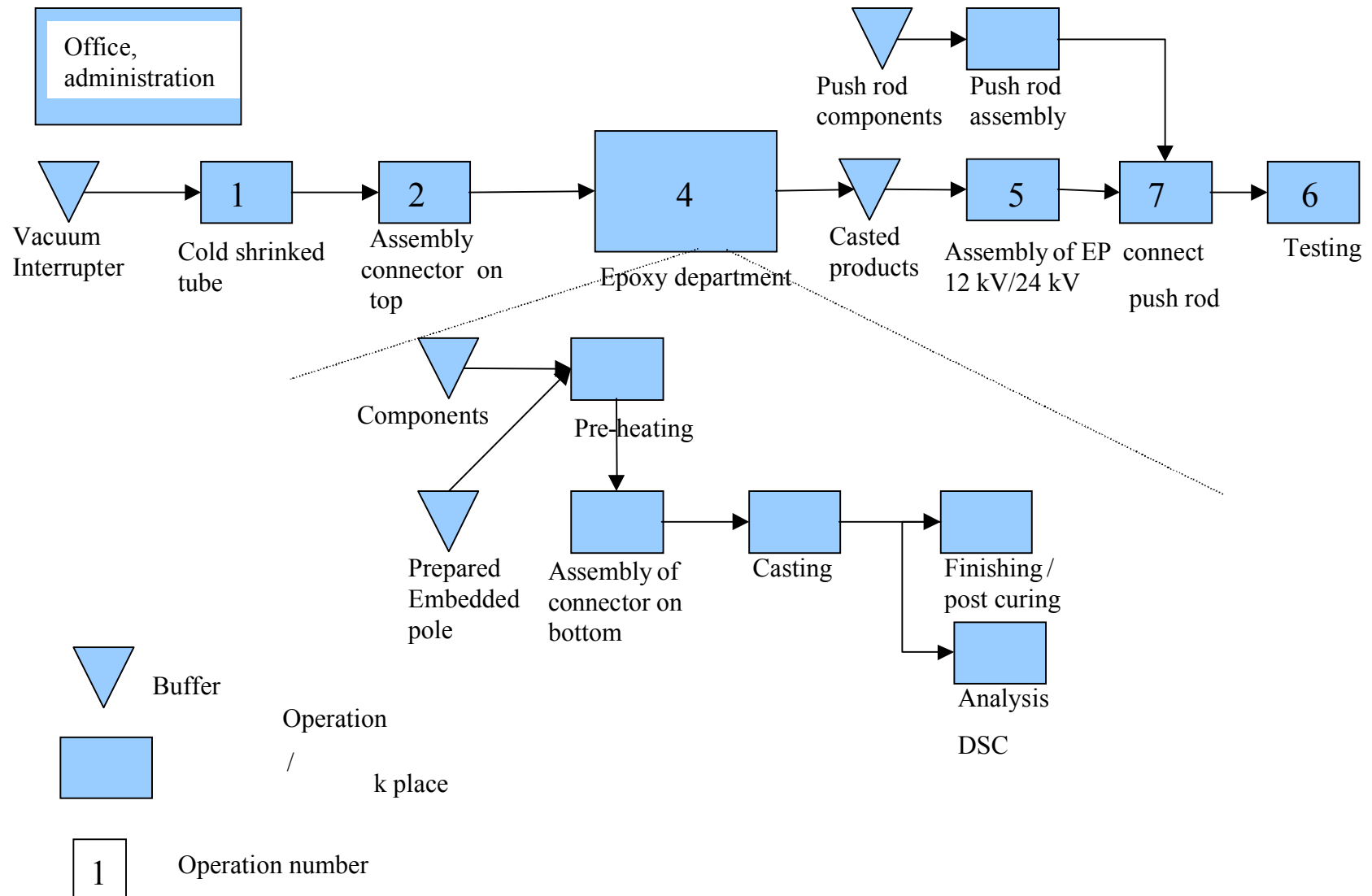


Example: manufacturing steps for switchgears

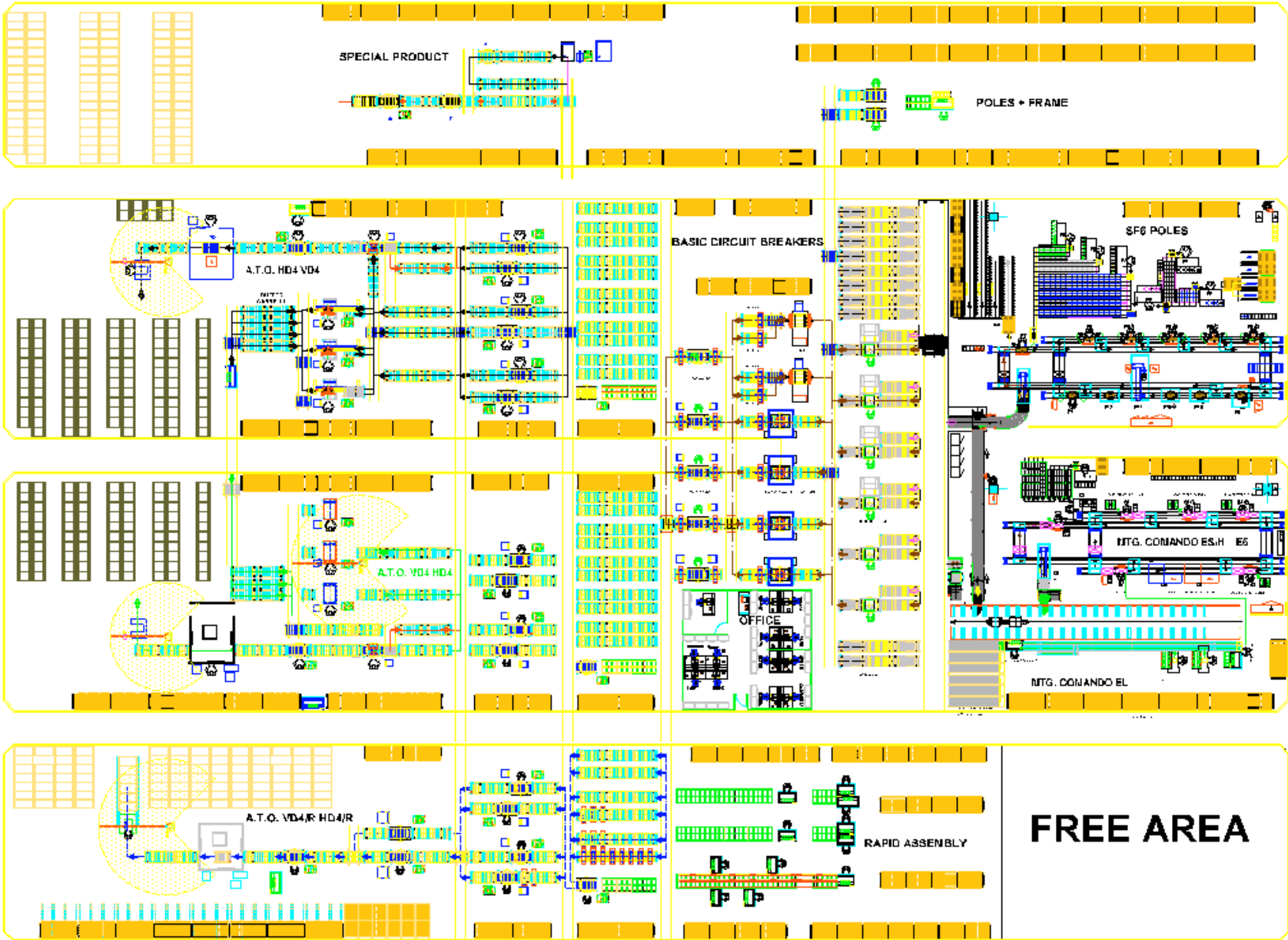


- Cold Shrink tube – Prepare the shrink tubes by labelling cold shrink tubes
- Assembly connector on top – assemble the connector on top of the 12 kV/24kV embedded poles, recording the torques.
- Assembly of high current EP – assemble, calculate quantity of washers, and record torque (only applies to high-current Eps)
- Epoxy Department – embed vacuum interrupter in epoxy in 4 steps: assembling connector to bottom, run the epoxy machinery, remove blur and analyse load number of resin
- Assembly of EP 12kV/24 kV assembly current strip and push rod, and record torques
- Testing - test continuous operation and voltage drops
- Assembly of push-rods – assembly according to part lists, spring force, and torque and generate barcode and print labels

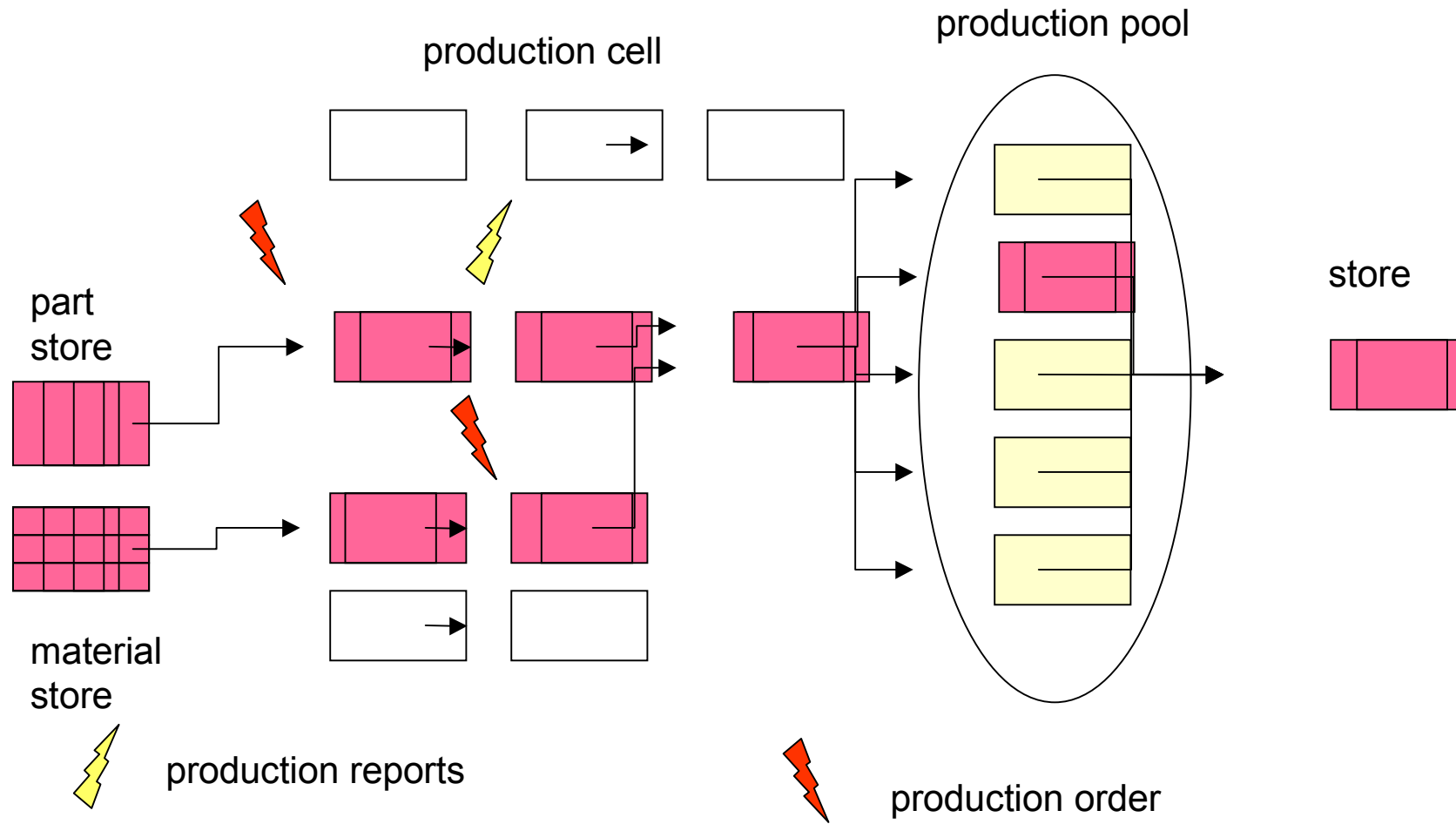
Example: Assembly process



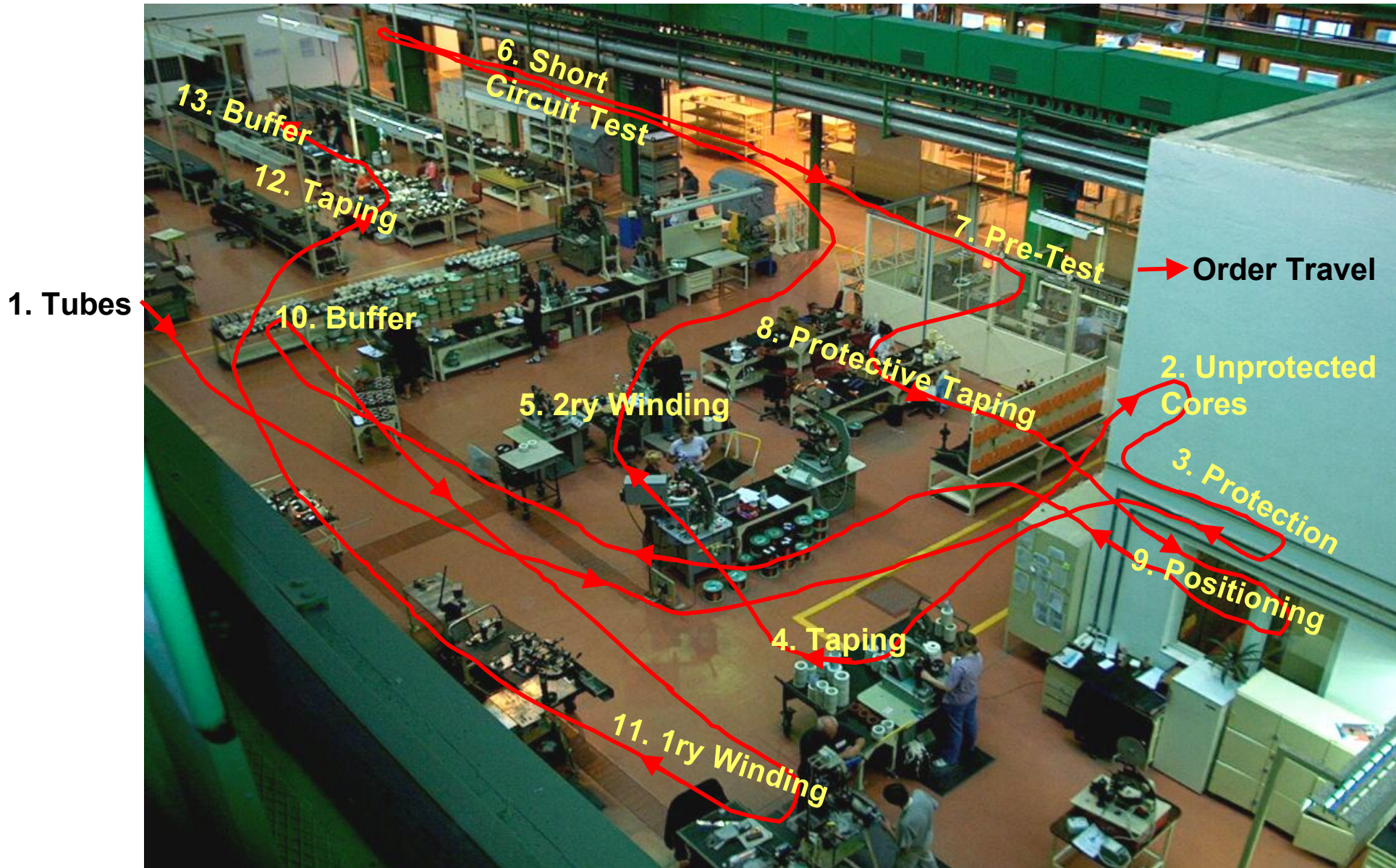
Example: Plant for manufacturing switchgears



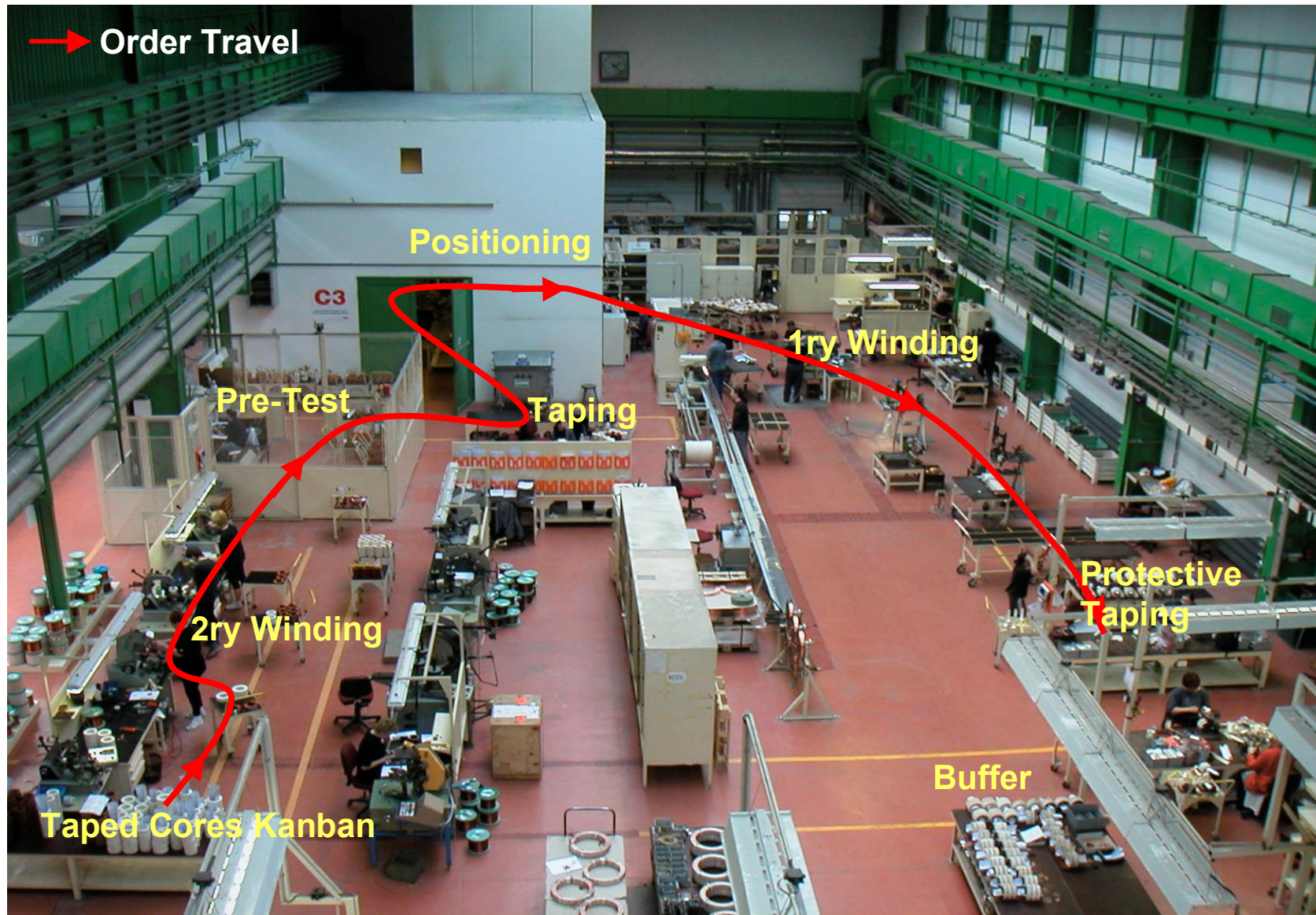
Dispatching and routing (workflow)



Workflow: Transportation, productivity and inventory waste ...

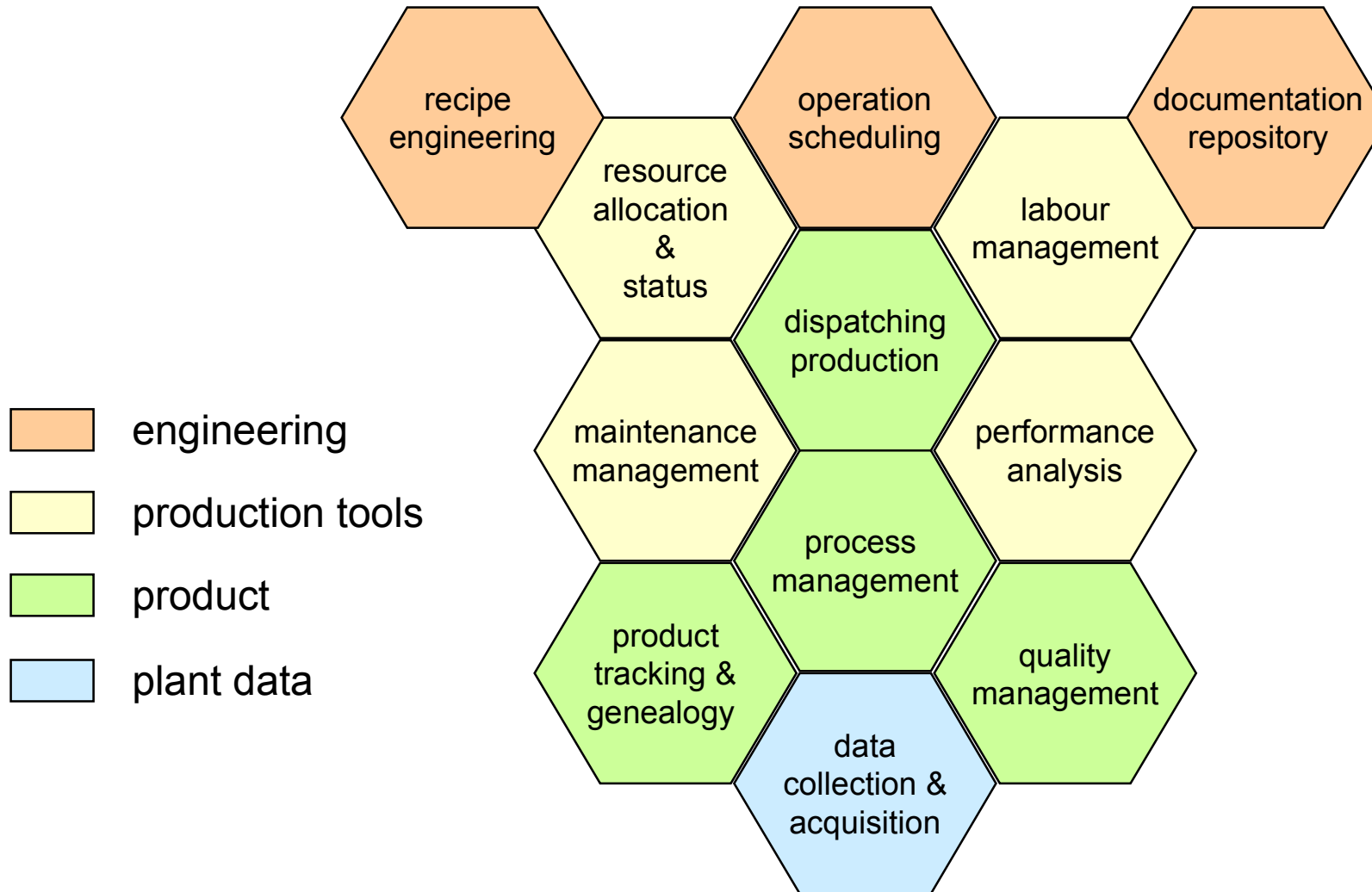


... have been vastly eliminated from the factory floor



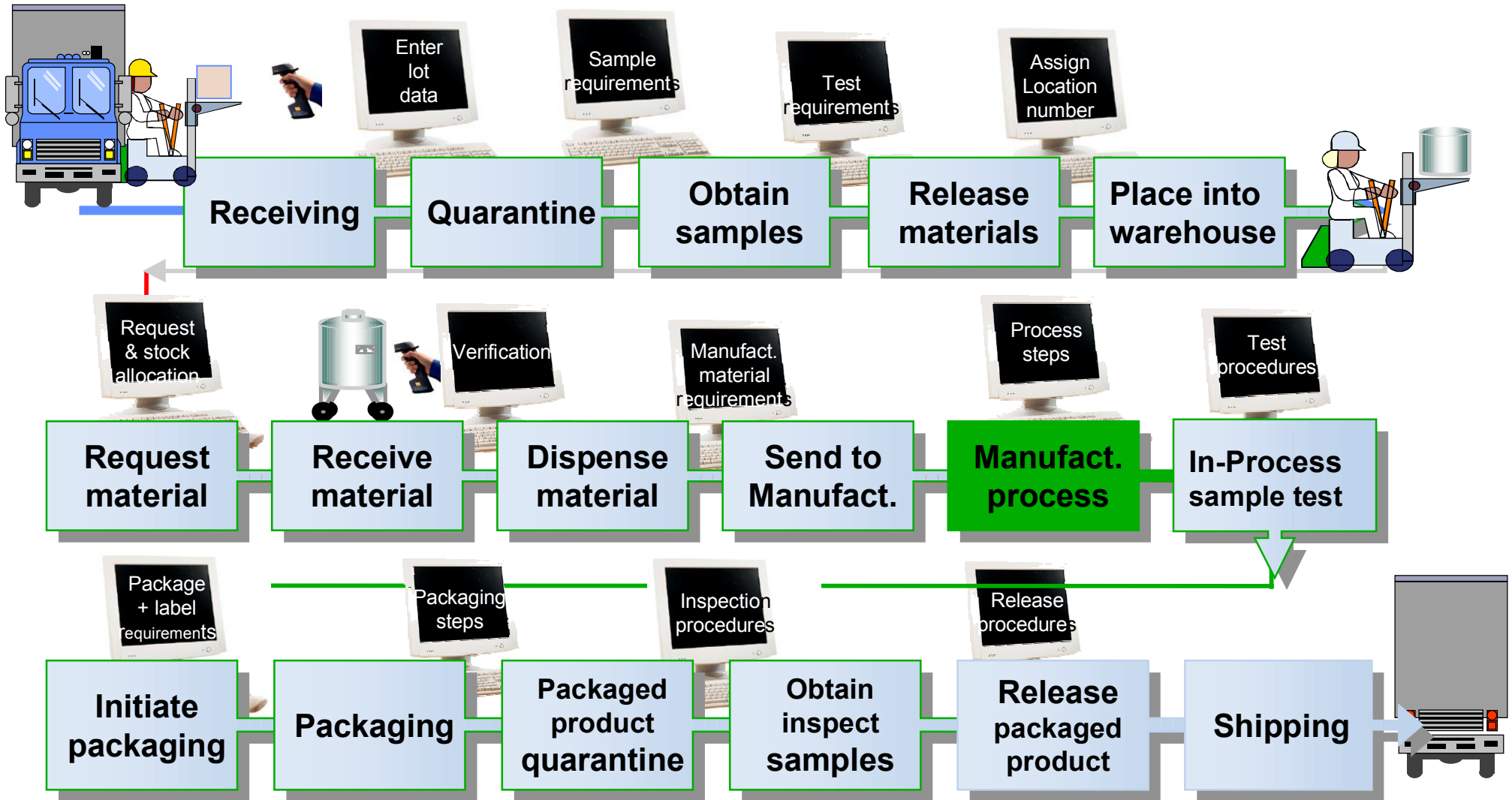
Level 3: Manufacturing Execution System

dispatch and control the manufacturing process based on actual (“real-time”) data



(ANSI/ISA 95 standard)

Manufacturing Workflow (e.g. pharmaceutical industry)



ISA S95: 1. Resource Allocation and Status

Guiding what people, machines, tools, and materials do, and what they are currently doing.

Maintains and displays status of resources including machines, tools, labour, materials, etc. that must be available in order for work to **start**.

Detail

- manage **resources** (machines, tools, labour skills, materials, other equipment, documents, ... that must be available for work to start and to be completed, directly associated with control and manufacturing.
- do local resource **reservation** to meet production-scheduling objectives.
- ensure that equipment is properly set up for processing, including any allocation needed for set-up.
- provide real-time **statuses** of the resources and a detailed history of resource use.

ISA S95: 2. Dispatching production (routing, workflow)

Giving commands to send materials or order to parts of the plant to begin a process or step.

Detail

- Manage the flow of production in the form of jobs, orders, batches, lots, and work orders, by **dispatching** production to specific equipment and personnel.
- Dispatch information is typically presented in the **sequence** in which the work needs to be done and may change in real time as events occur on the factory floor.
- Alter the prescribed schedules, within agreed upon limits, based on local availability and current conditions.
- Control the amount of work in process at any point through buffer management and management of **rework** and **salvage** processes.

ISA S95: 3. Data Collection

Monitoring, gathering, and organizing data about processes, materials, and operations from people, machines, or controls.

Ability to collect and store data from production systems to use for population of forms and records. Data can be collected manually or automatically in real time increments

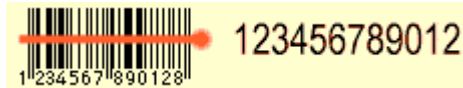
Detail

- obtain the operational production and parametric data associated with the production equipment and processes.
- provide real-time status of equipment and production processes and a history of production and parametric data.

3. Data Collection Input devices specific for manufacturing

EAN Barcode

universal input device, serial number, error report.
Limited text length



Bar code label printer



Bar code scanner



PDF417: upcoming standard, high density coding
even small ink quantities may impair some products.

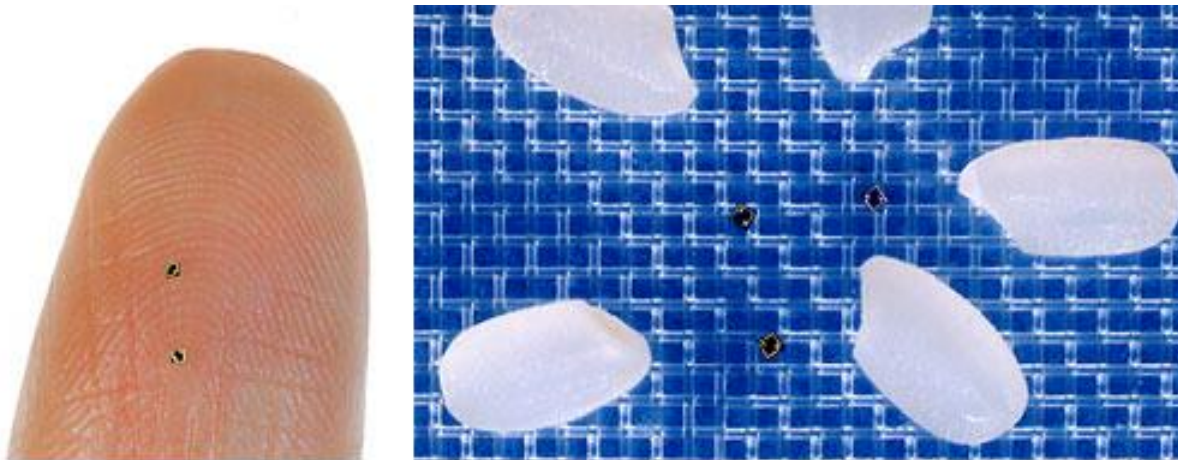
3. Data Collection RFIDs

RFID = Radio Frequency Identifiers

Hundreds or even thousands of tags can be identified at the same time at distance of 3m with a single reader antenna and 6m between two reader antennas.

At 13.56 MHz can store 512 bits, new versions working in the 915 MHz range
Price: 0.1 € / piece

Unsuitable on metal, high temperatures, - for the better and the worse.



A New RFID with Embedded Antenna μ -Chip

3. Data Collection Local HMI

Workorders

Priority	Work Order	Product Family
1	401298	I.VD4
2	401299	I.HD4/R

Quantity of order: 10 Quantity done: 2

Procedures

Control on the frame that the threads of the nuts and hinges aren't painted
Controllare che i perni dei gruppi biella siano ingrassati

Pole Configuration

Control visually whether the type of the pole corresponds to the required one:

- VG5 small pole
- VG4 large pole with black ring
- VG4S large pole with red ring

Configuration checked

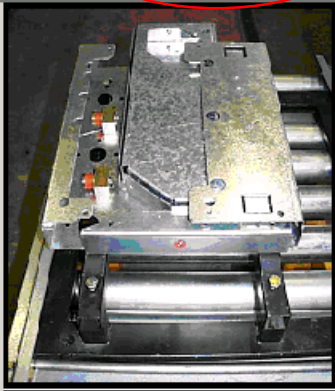
Contrassegnare con pennarello le viti di fissaggio dei gruppi biella alla struttura
Incollare sulla fiancata sx, all'interno della struttura, il barcode

Statics

Description: INTERRUTTORE HD4/R 24.06.16 16 P320
OR AKN: 1602006442
Position of CO: 000100
Customer Description: ICIE SRL
PO Delivery Date: 04.12.2002
CO Delivery Date: 05.12.2002
Serial Number Start: AD000A6228
Serial Number End: AD000A6328

Pallet Identification


Support Tool ID: 200
Serial Number: AD000A6230



Supervisor Message

5/3/2003 13:53

DPMO



Quick Info

Work Instructions

Operating Instructions

Tool Verification

Maintenance

Bill Of Material

Safety

Drawing

SPC

History

Call Supervisor

Call Quality

Call Maintenance

Logout

Done Suspend Cancel Repair

ISA S95: 4. Quality Management

Recording, tracking and analyzing product/process characteristics against engineering needs.

Detail

- provide real-time measurements collected from manufacturing and analysis in order to assure proper product quality control and to identify problems requiring attention.
- Recommend corrections, including correlating the symptoms, actions and results to determine the cause.
- **SPC/SQC** (statistical process control/statistical quality control) tracking and management of offline inspection operations and analysis in laboratory information management systems (LIMS).

4: Quality Test

STEPS in assembly:

- Scan serial # from cabinet to id unit
- Examine Work Order
- Package both cabinets for shipping
- **Fill out checklist & test reports**
- Update Syteline & ship

Pack next cabinet



ABB DISTRIBUTION AUTOMATION EQUIPMENT DIVISION
LAKE MARY, FLORIDA
CERTIFIED TEST REPORT - RETROFIT CABINETS

GENERAL ORDER # _____ SHOP ORDER _____

UNIT SERIAL # _____ CUSTOMER # _____

PCD STYLE # _____ PCD SERIAL # _____

SOFTWARE VERSION NUMBER _____

FRONT PANEL CONTROLS

- A. REMOTE ENABLE _____ OK
 B. GROUND BLOCK _____ OK
 C. ALTERNATE PU _____ OK
 D. SEF ENABLE _____ OK (WHEN APPLICABLE)
 E. RECLOSE BLOCK _____ OK
 F. PROG. 1 _____ OK (BATTERY TEST)
 G. FAULT TEST _____ OK (SELF TEST)

CONTROL FUNCTIONS

- A. MINIMUM PICKUP, PHASE 1 _____ OK PHASE 2 _____ OK PHASE 3 _____ OK GROUND _____ OK
 B. INSTANTANEOUS TRIPPING _____ OK
 C. TIME DELAY TRIPPING _____ OK
 D. RECLOSE TIMES _____ OK
 E. RESET TIME _____ OK

INPUT/OUTPUT TEST

INTERLOCKED WITH REOTE ENABLED FUNCTION

- REMOIST CLOSE _____ OK
 REMOTE TRIP _____ OK
 REMOTE RECLOSE BLOCK _____ OK
 REMOTE ALT. 1 _____ OK

INDEPENDENT OF REMOTE ENABLE FUNCTION

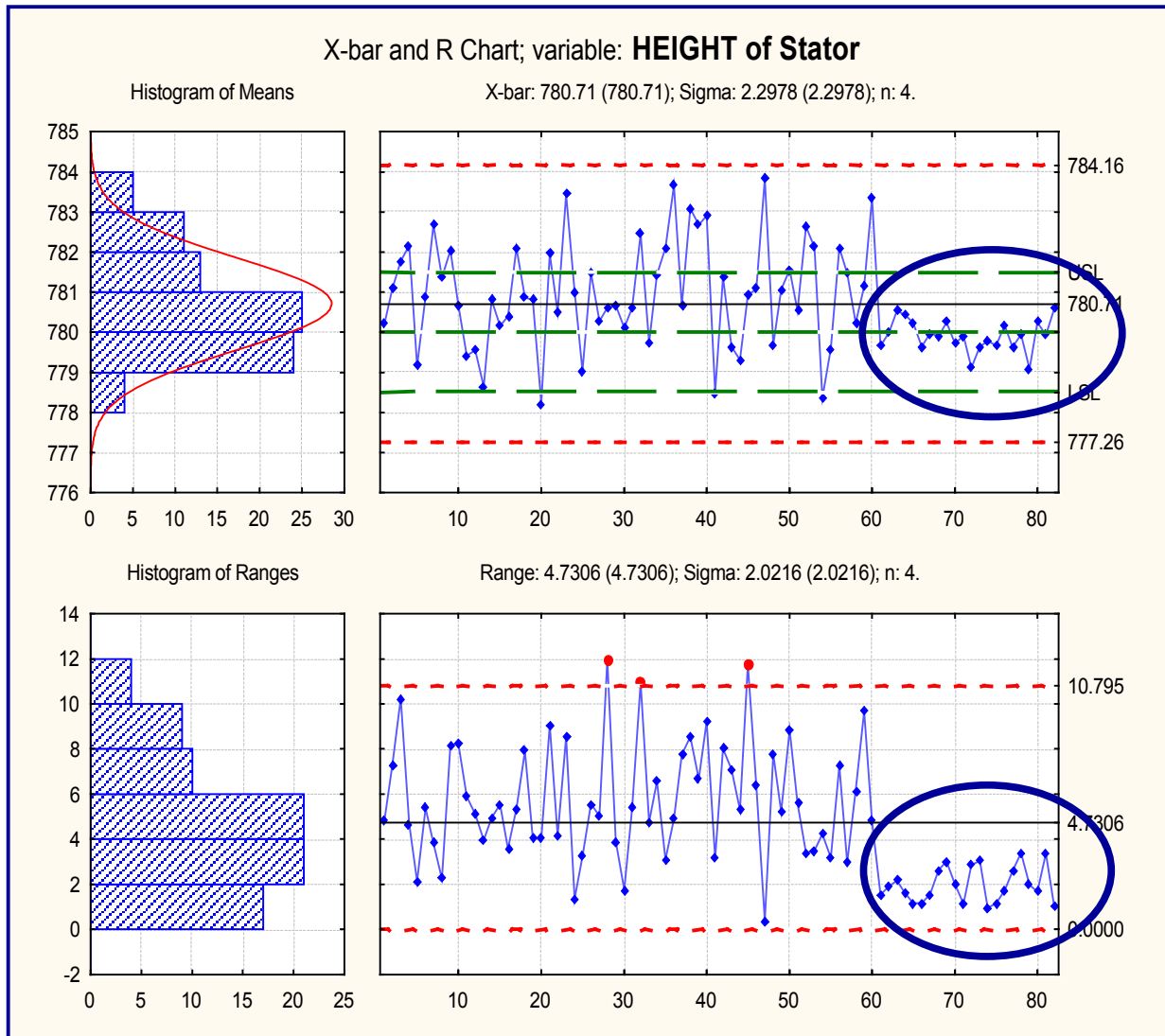
- SUPERVISOR CLOSE _____ OK
 SUPERVISORY TRIP _____ OK

VOLTAGE WITHSTAND

CHECK THE CONTROL CABINET WIRING, TO GROUND, AT 1500 VAC FOR

Typical Final Inspection Checklist

4: Example of quality statistics



ISA S95: 5. Process Management

Directing the flow of work in the plant based on planned and actual production activities.

Detail

- monitor production and either automatically corrects or provides decision support to operators for correcting and improving in-process functions.
These functions may be intra-operational and focus specifically on machines or equipment being monitored and controlled, as well as inter-operational, tracking the process from one operation to the next.
- manage **alarms** to ensure factory persons are aware of process changes that are outside acceptable tolerances.

ISA S95: 6. Product Tracking & Genealogy

Monitoring the progress of units, batches, or lots of output to create a full product history.

Detail

- Monitors and tracks material used in a manufactured part including revisions, sources, serial numbers, supplier identification, or lot.
This information is retrievable in the event of quality problems or process changes to identify comparable products.
- record information to allow forward and backward traceability of components and their use within each end product.

ISA S95: 7. Performance Analysis

Comparing measured results in the plant to goals and metrics.

Ability to consolidate collected data and calculate results including real production cost, uptime, SPC/SQC of production parts, etc. Includes comparison of current vs. historical performance.

Detail

- Provide up-to-the-minute reporting of actual manufacturing operations results along with comparisons to past history and expected results.
- Performance results include such measurements as resource utilization, resource availability, product unit cycle time, conformance to schedule, and performance to standards.
- Include SPC/SQC analysis and may draw from information gathered by different control functions that measure operating parameters.

7. Performance Analysis: questions the factory owner asks

What is the number of good / bad pieces produced: by shift X, in week 20 ?
(with / without induced downtime)
What is the relation to the maximum ?

What was the average production speed of a unit compared to the maximum ?
What is the production speed in function of time, deducing stops ?

How much afar from the theoretical production capacity is my plant producing ?

What are the N major reasons why the unit is not producing at full capacity ?
How many stops did the unit suffered ?

What is the availability of my production unit

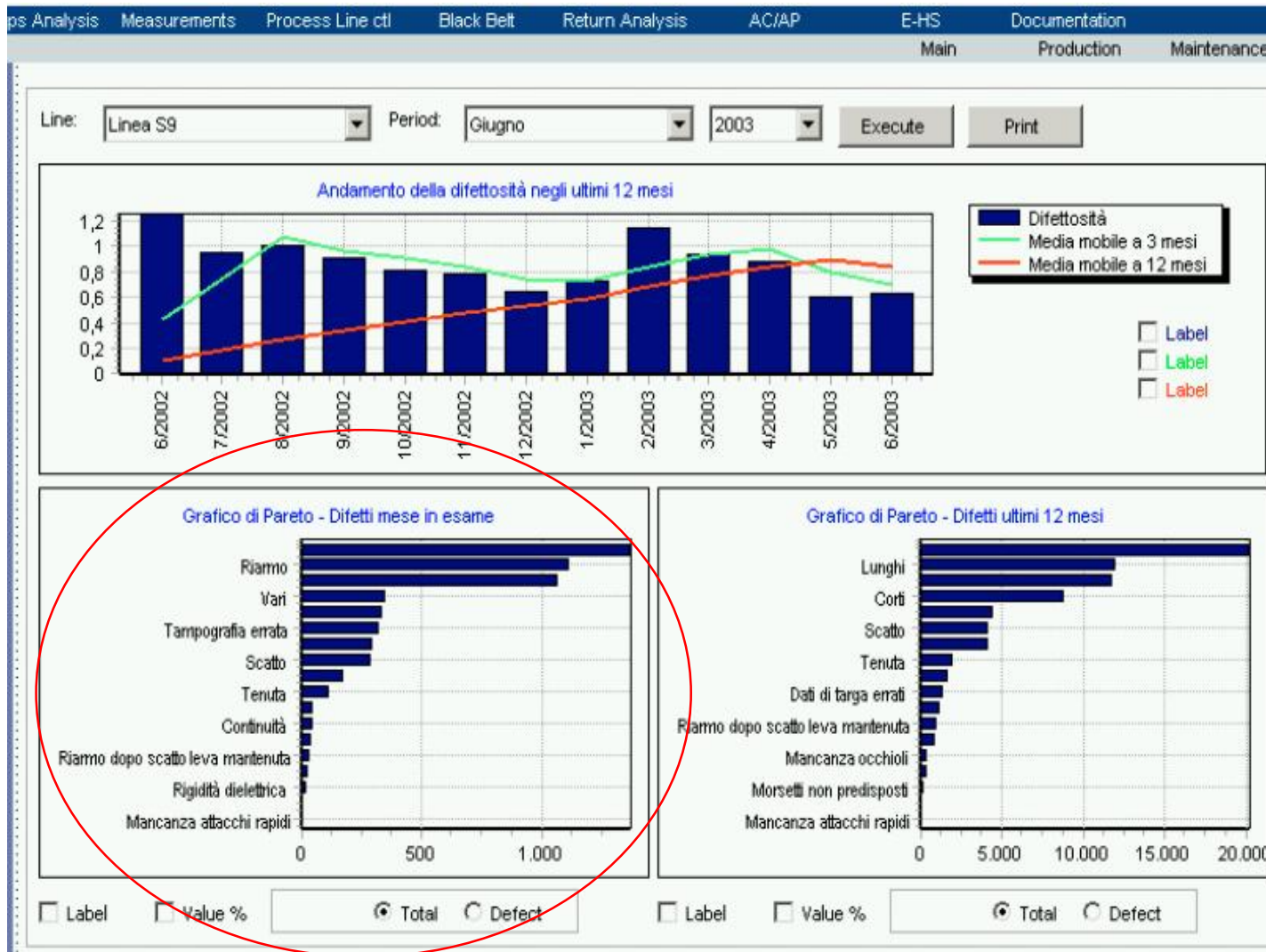
What is the efficiency of operator M ?, of shift S ?

What is the progression of the OEE (overall equipment efficiency) on a daily basis ?

How much time is spent loading / unloading the machine ?

How does my OEE compares with others ?

7. Performance analysis and Pareto



ISA S95: 8. Operations and detailed scheduling

Sequencing and timing activities for optimised plant performance based on finite capacities of the resources

Detail

- Provide **sequencing** based on priorities, attributes, characteristics, and production rules associated with specific production equipment and specific product characteristics, such as shape, colour sequencing or other characteristics that, when scheduled in sequence properly, minimize set-up.
- Operations and detailed scheduling is finite and it recognizes alternative and overlapping/parallel operations in order to calculate in detail the exact time of equipment loading and adjustment to shift patterns.

ISA S95: 9 Document Control

Managing and distributing information on products, processes, designs, or orders.

Controls records and forms that must be maintained to serve regulatory and quality needs and populates those forms with actual production data.

Also maintains current documents provided to operators to assist in production methods.

Detail:

- control records and forms that must be maintained with the production unit. (records and forms include work instructions, recipes, drawings, standard operation procedures, part programs, batch records, engineering change notices, shift-to-shift communication, as well as the ability to edit "as planned" and "as built" information).
- send instructions down to the operations, including providing data to operators or recipes to device controls.
- control and integrity of regulatory, documentation, environmental, health and safety regulations, and operative information such as corrective action procedures.

SA S95: 10 Labour Management

Tracking and directing the use of operations personnel based on qualifications, work patterns and business needs

detail

- provide status of personnel in an up-to-the minute time frame.
- provide time and attendance reporting, certification tracking,
- track indirect functions such as material preparation or tool room work as a basis for activity-based costing.
- interact with resource allocation to determine optimal assignments.

ISA S95: 11. Maintenance Management

Planning and executing activities to keep capital assets in the plant performing to goal.

Detail

- Maintain equipment and tools.
- Ensure the equipment and tools availability for manufacturing.
- Schedule periodic or preventive maintenance as well as responding to immediate problems.
- Maintain a history of past events or problems to aid in diagnosing problems.

Additional definitions

12. Work order tracking (not S95)

Directing the flow of work in the plant based on planned and actual production activities

Monitors work orders as they pass through the operations. Real time status provides management with view of actual production output and permits workflow changes based on business rules.

13. Recipe Manager: (not S95)

Mapping production order operations to detailed list of tasks/jobs, providing detailed recipe for manufacturing

Conclusion

MES is a business of its own, that require a good knowledge of the manufacturing process and organization skills.

Simulation tools are helpful to anticipate the real plant behavior

Although buzzwords abound (“lean manufacturing”,....), it is more an issue of common sense than of science.

Assessment

Which are the parts of the ISA S95 standard ?

What does Kanban means ?

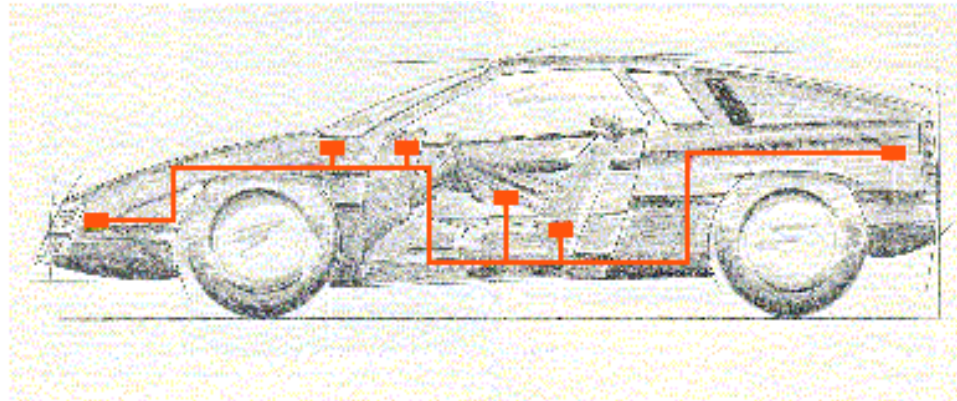
What is asset management ?

Which manufacturing models exist ?

What is a KPI and which KPI is a client interested in ?

Which level 1 plants does ISA S95 consider ?





8

Real-time consideration
Considération du temps réel
Echtzeit - Berücksichtigung

Prof. Dr. H. Kirrmann

ABB Research Center, Baden, Switzerland

Real-time constraints

Marketing calls "real-time" anything "fast", "actual" or "on-line"

Definition: A real-time control system is required to produce output variables that respect defined time constraints.

Levels of real-time requirements:

- meet all time constraints exactly (hard real-time)
- meet timing constraints most of the time (soft real-time)
- meet some timing constraints exactly and others mostly.

These constraints must be met also under certain error conditions

Effects of delays

- In regulation tasks, delays of the computer appear as dead times, which additionally may be affected by jitter (variable delay).
- In sequential tasks, delays slow down plant operation, possibly beyond what the plant may tolerate.

Reaction times

- 10 μ s: positioning of cylinder in offset printing (0,1 mm at 20 m/s)
- 46 μ s: sensor synchronization in bus-bar protection for substations (1° @ 60Hz)
- 100 μ s: resolution of clock for a high-speed vehicle (1m at 360 km/h)
- 100 μ s: resolution of events in an electrical grid
- 1,6 ms: sampling rate for protection algorithms in a substation
- 10 ms: resolution of events in the processing industry
- 20 ms: time to close or open a high current breaker
- 200 ms: acceptable reaction to an operator's command (hard-wire feel)
- 1 s: acceptable refresh rate for the data on the operator's screen
- 3 s: acceptable set-up time for a new picture on the operator's screen
- 10 s: acceptable recovery time in case of breakdown of the supervisory computer
- 1 min: general query for refreshing the process data base in case of major crash

Processing times

- 1 μ s: addition of two variables in a programmable logic controller
- 10 μ s: execution of an iteration step for a PID control algorithm.
- 30 μ s: back- and forth delay in a 3'000 m long communication line.
- 40 μ s: coroutine (thread) switch within a process
- 160 μ s: send a request and receive an immediate answer in a field bus
- 100 μ s: task switch in a real-time kernel
- 200 μ s: access an object in a fast process database (in RAM)
- 1 ms: execution of a basic communication function between tasks
- 2 ms: sending a datagram through a local area network (without arbitration)
- 16 ms: cycle time of a field bus (refresh rate for periodic data)
- 60 ms: cycle time of the communication task in a programmable logic controller.
- 120 ms: execution of a remote procedure call (DCOM, CORBA).

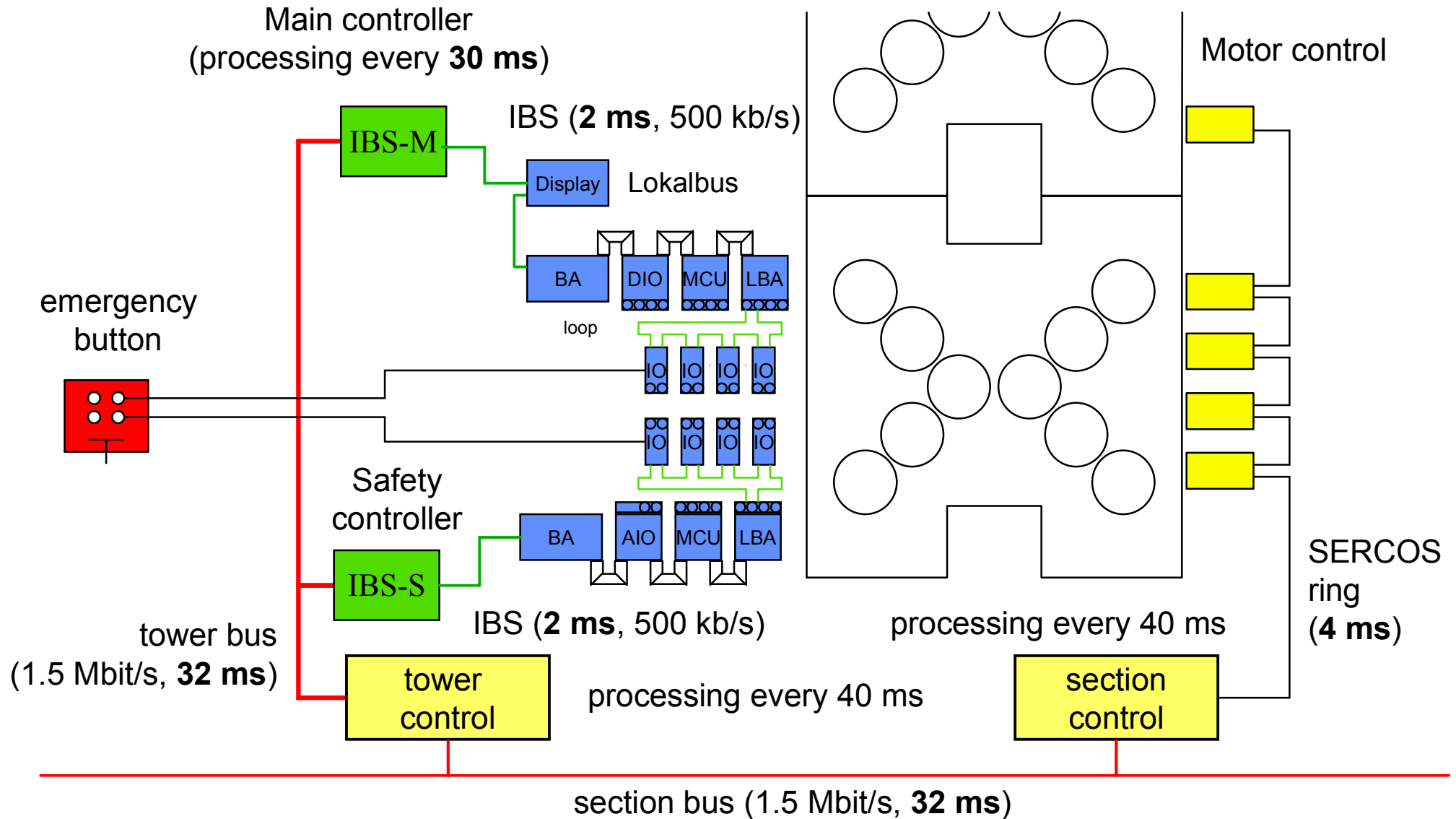
What real-time response really means

Emergency stop



The operator keep one hand on the “rotate” button while he washes with the other.
If the towel gets caught, he releases the button and expects the cylinder to stop in 1/2 second ...

The signal path from the emergency stop to the motor



Total delay path: $2 + 30 + 32 + 40 + 32 + 40 + 4 = 180 \text{ ms} !$

Delay path and reaction time

Most safety systems operate negatively:

-> lack of “ok” signal (life-sign toggle) triggers emergency shutdown

The motor control expects that the information “emergency button not pressed” is refreshed every $3 \times 180 = 540$ ms to deal with two successive transmission errors, otherwise it brakes the motors to standstill.

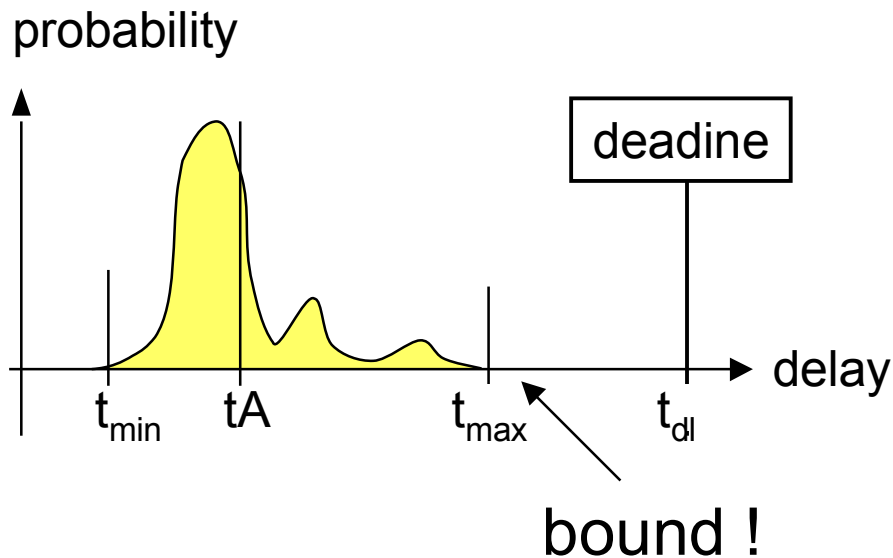
Excessive signal delay causes false alarms -> affects availability of the plant
(client won't accept more than 1-2 emergency shutdown due to false alarm per year)

Therefore, control of signal delays is important:

- for safety
- for availability

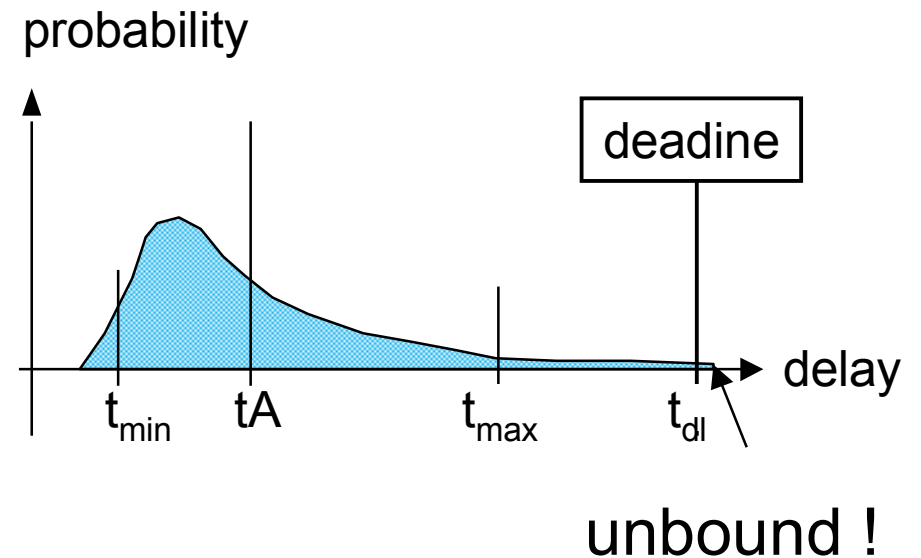
Hard- and Soft real time

hard real-time
(deterministic)



the probability of the delay to exceed an arbitrary value is zero under normal operating conditions, including recovery from error conditions

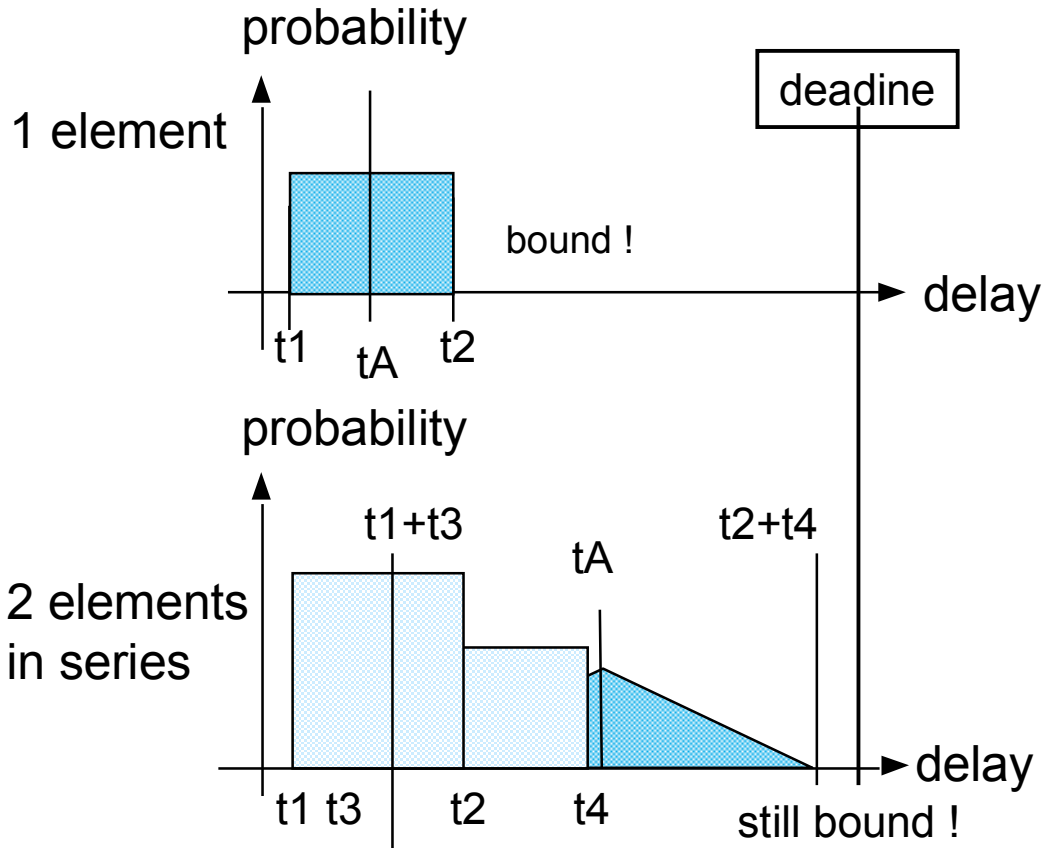
soft real-time
(non-deterministic)



the probability of the delay to exceed an arbitrary value is small, but non-zero under normal operating conditions, including recovery from error conditions

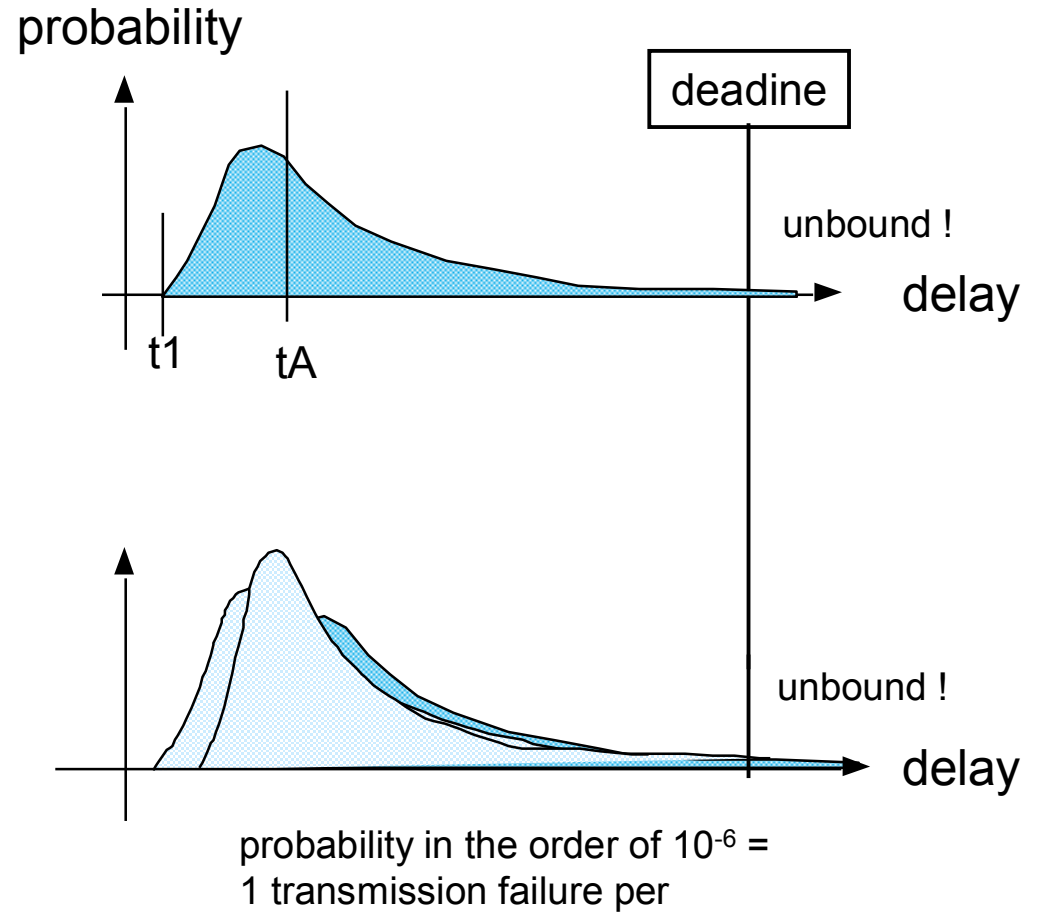
Hard Real-Time and Soft Real-Time: series connection

hard real-time
(cyclic)

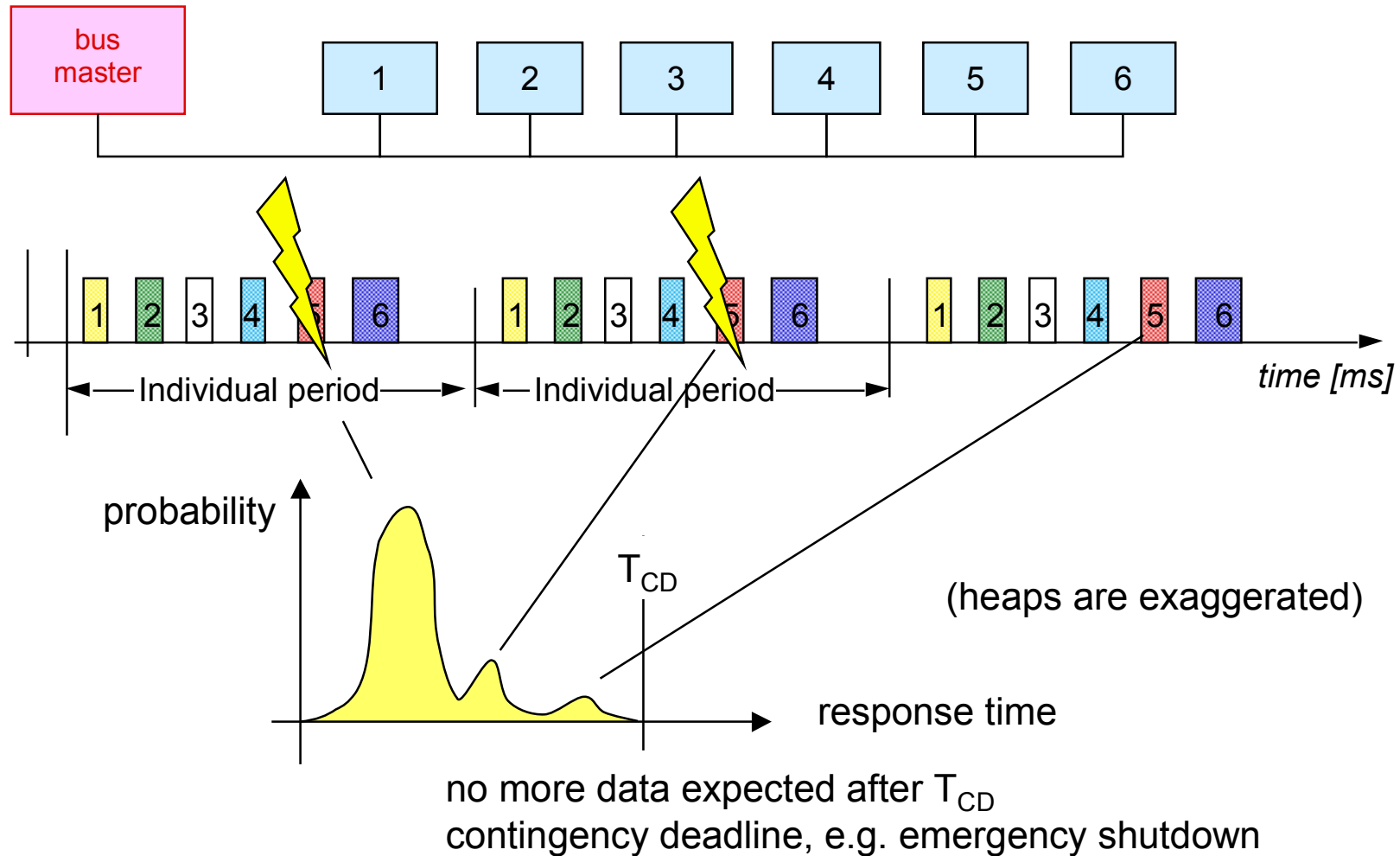


probability of two elements in series = convolution integral

soft real-time
(event-driven, CSMA)



Determinism and transmission failures



Example: probability of data loss per period = 0.001,
probability of not meeting T_{CD} after three trials = 10^{-9} ,
same order of magnitude as hardware errors -> emergency action is justified.

Deterministic systems

A deterministic system will react within bound delay under all conditions.

A deterministic system can be defeated by external causes (failure of a device, severing of communication line), but this is considered as an accepted exceptional situation for which reaction is foreseen.

Determinism implies previous reservation of all resources (bus, memory space,...) needed to complete the task timely.

All elements of the chain from the sensor to the actor must be deterministic for the whole to behave deterministically.

Non-deterministic components may be used, provided they are properly encapsulated, so their non-determinism does not appear anymore to their user.

Examples:

- queues may be used provided:
 - a high-level algorithm observed by all producers ensures that the queues never contains more than N items.
- Interrupts may be used provided:
 - the interrupt handler is so short that it may not cause the interrupted task to miss its deadline, the frequency of interrupts being bound by other rules (e.g. a task has to poll the interrupts)

Deterministic Control Systems

For real-time systems, small, affordable and well-understood kernels are used: VRTX, VxWorks, RTOS, etc....

The tasks in these systems normally operate cyclically, but leave room for event processing when idle - the cyclic task must always be able to resume on time.

Control network does not depend on raw speed, but on response time.

Control loops need timely transmission of all critical variables to all sink applications.

If an application sends one variable in 7 ms to another application, transmission of all variables may require $n \times 7$ ms (except if several variables are packed in one message).

If several applications are interested in a variable, the number of transfer increases, except if transmission is (unacknowledged) broadcast.

Smooth execution of control algorithms require that data are never obsolete by more than a certain amount.

Determinism is closely related to the principle of cyclic operation

Non-deterministic systems

A non-deterministic system can fail to meet its deadline because of internal causes (congestion, waiting on resource), without any external cause.

Computers and communication may introduce non-deterministic delays, due to internal and external causes:

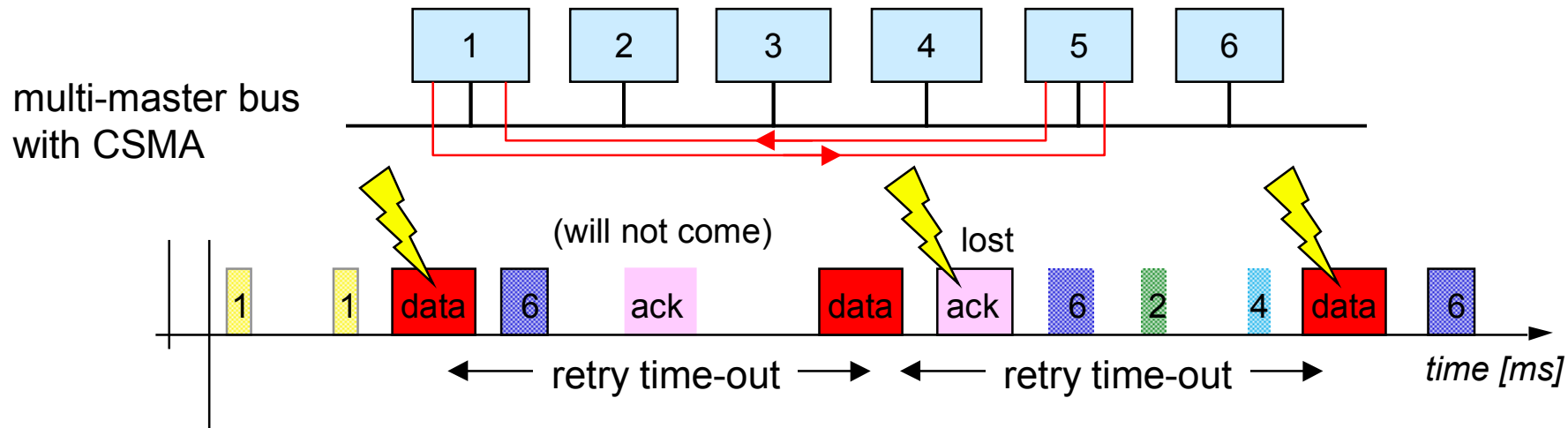
- response to asynchronous events from the outside world (interrupts)
- access to shared resources: computing power, memory, network driver,...
- use of devices with non-deterministic behavior (hard-disk sector position)

Non-determinism is especially caused by:

- Operating system with preemptive scheduling (UNIX, Windows,..) or virtual memory (in addition, their scheduling algorithm is not parametrizable)
- Programming languages with garbage collection (Java, C#, ...)
- Communication systems using a shared medium with collision (Ethernet)
- Queues for access to the network (ports, sockets)

Non-determinism is closely related to on-demand (event-driven) operation

Failures in Ethernet - Style transmission



Probability of transmission failure due to collision: e.g. 1% (generous)
 (Note: data loss due to collision is much higher than due to noise !)

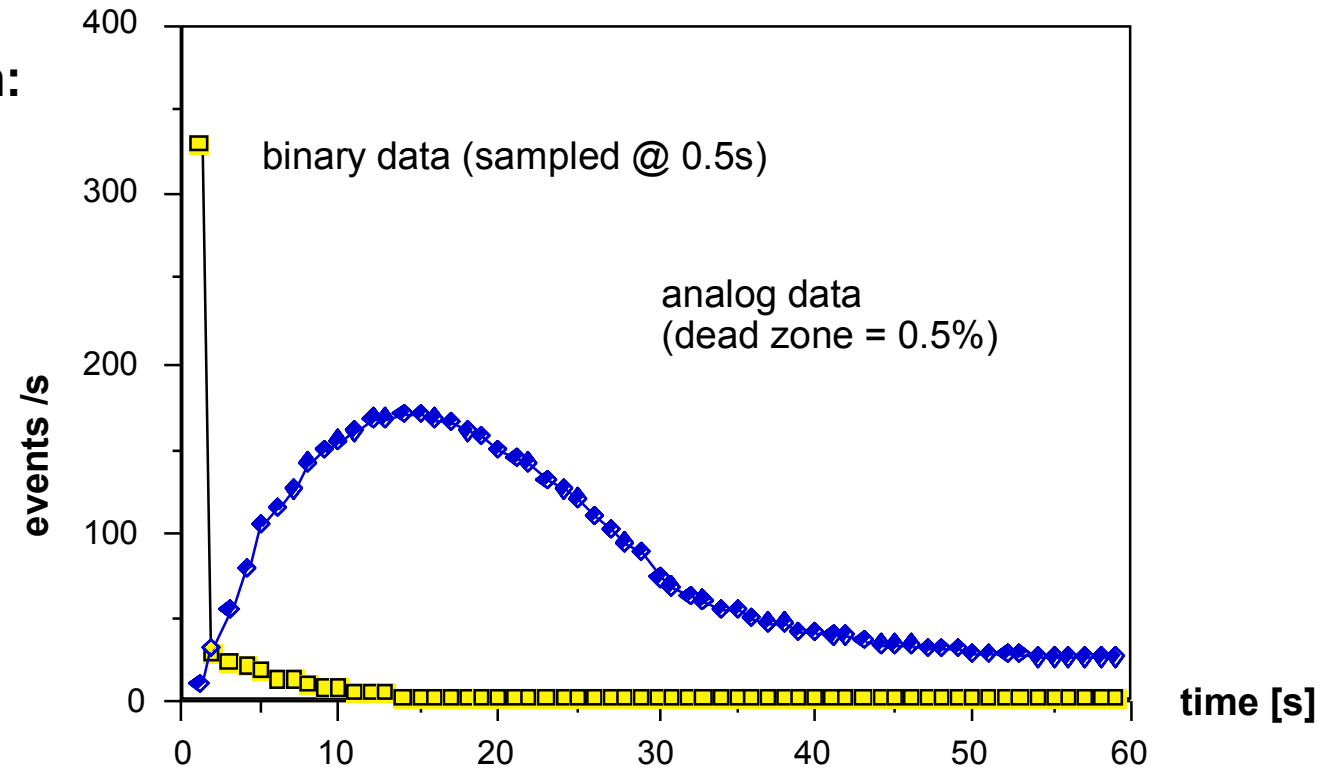
With no collision detection, retransmission is triggered by not receiving acknowledgement of remote party within a time T_{rto} (reply time-out).

This time must be larger than the double queue length at the sender and at the receiver, taking into account bus traffic. Order of magnitude: 100 ms.

The probability of missing three T_{rto} in series is G^3 times larger than a cyclic system with a period of 100 ms, G being the ratio of failures caused by noise to failures caused by collisions (here: 1% vs. 0.01% $\rightarrow 10^6$ more emergency stops).

Case study: Analysis of the response of an event-driven control system

Typical stress situation:
loss of power



Binary variables: event is a change of state

Analog variables: event is a change of value by more than 0.5 %

Solution 1: PLC attach to plant through Field Bus

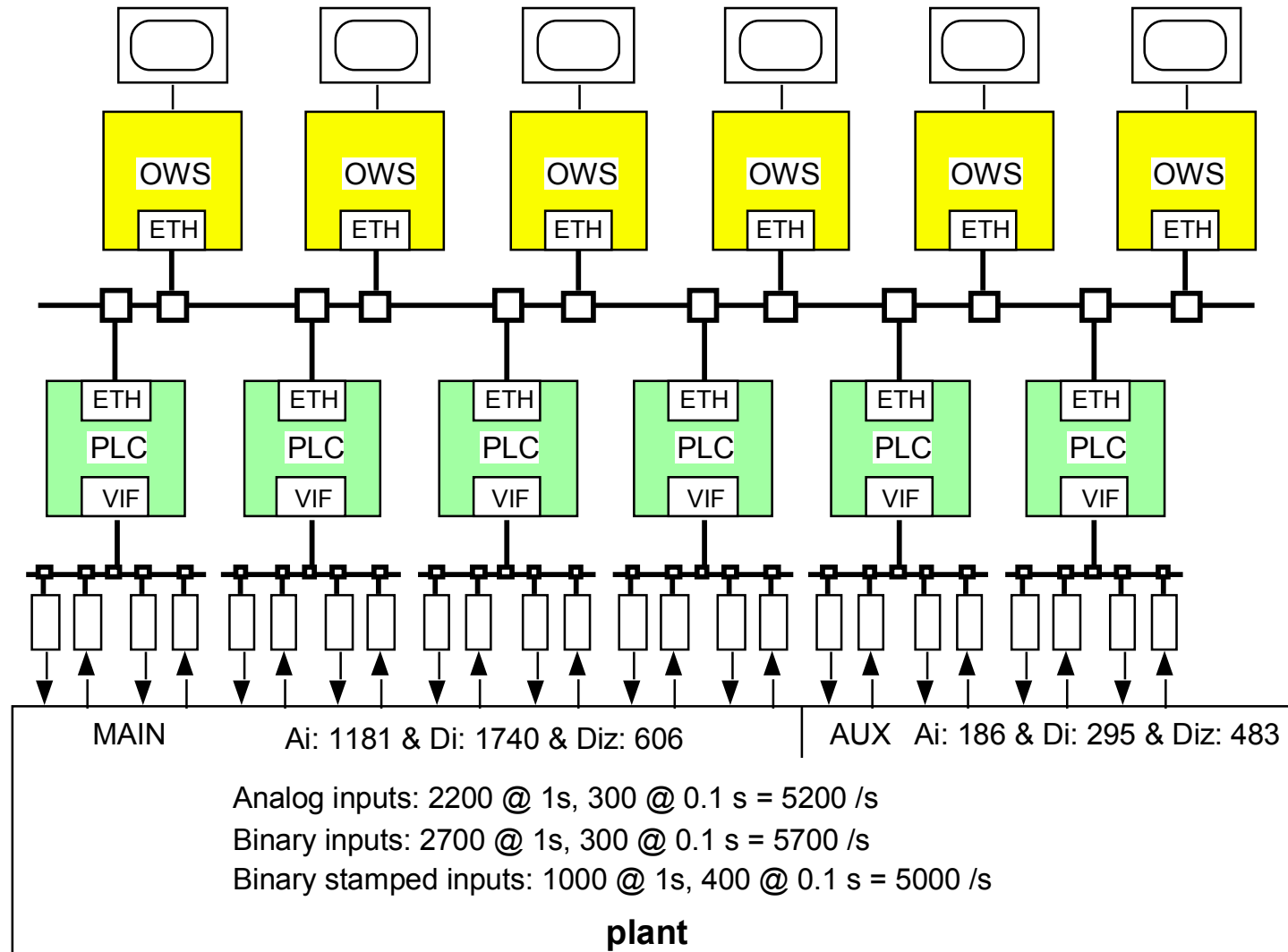
Up to 40
Operator
Workstation
1000 events/s
each

Ethernet
12'500 events/s
@ 10% load

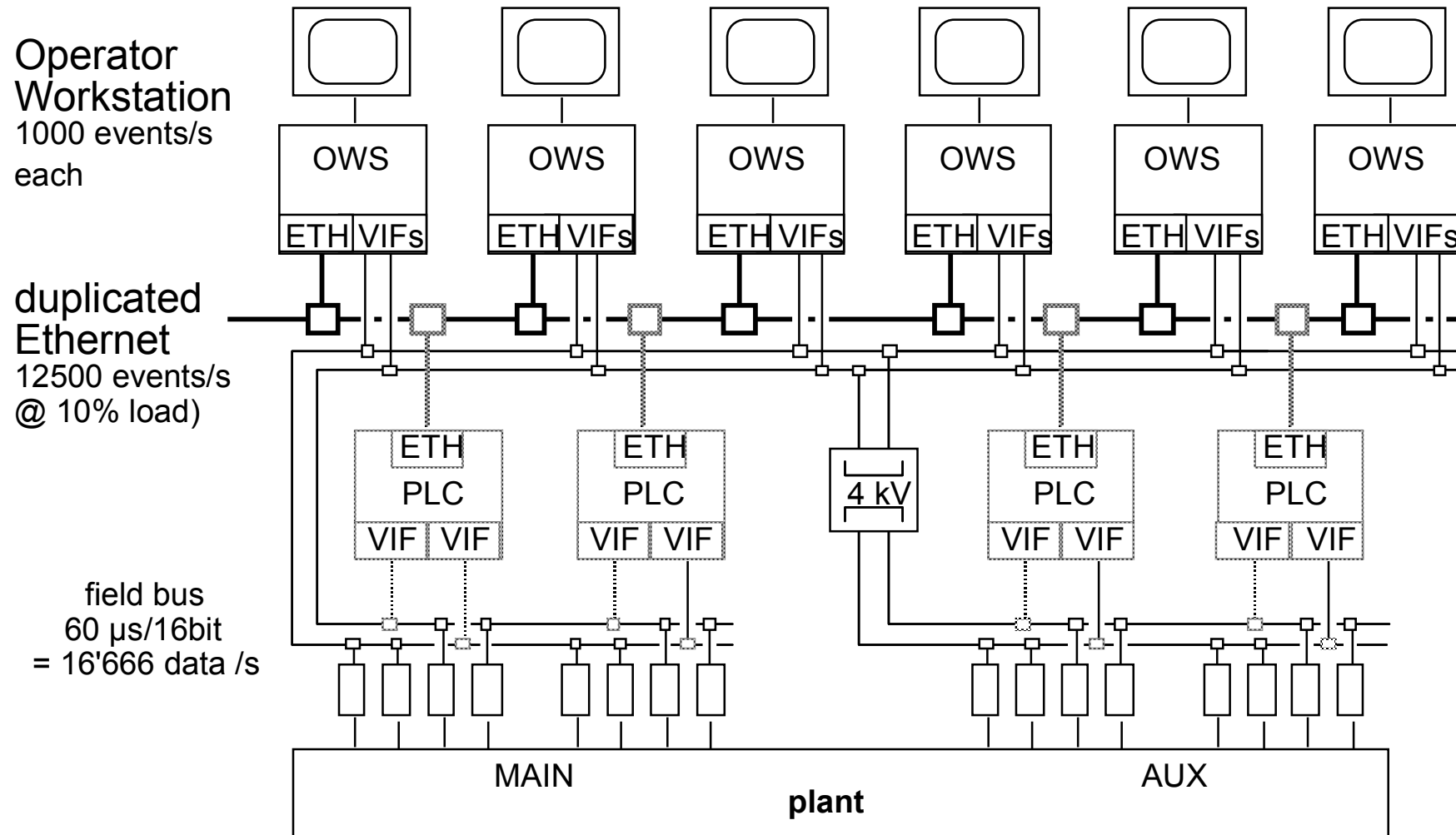
up to 6 PLC
300 events/s
each

Field Busses
60 μ s/16bit
= 16'666 data /s

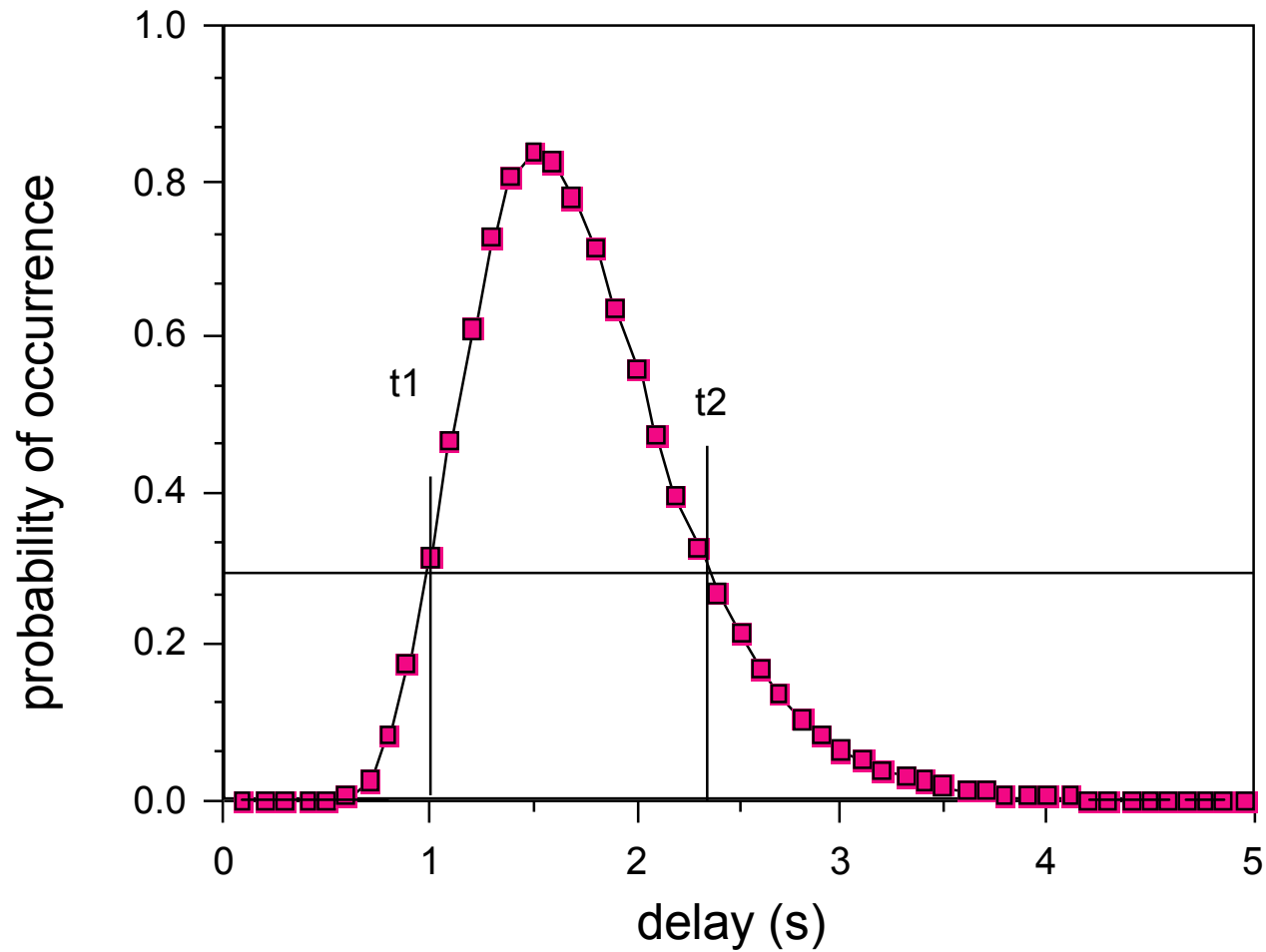
Total : 15'900 samples/s



Solution 2: OWS access Field Bus and PLCs directly



Event Processing: delay until a changed variable is displayed



The analysis of the delay distribution in all possible cases requires a complete knowledge of the plant and of the events which affect the plant.

It is not only event transmission which takes time, but also further processing

What is the worst-case condition ?

Every second, 15'900 variables are sampled, but most of them do not change and do not give rise to an event.

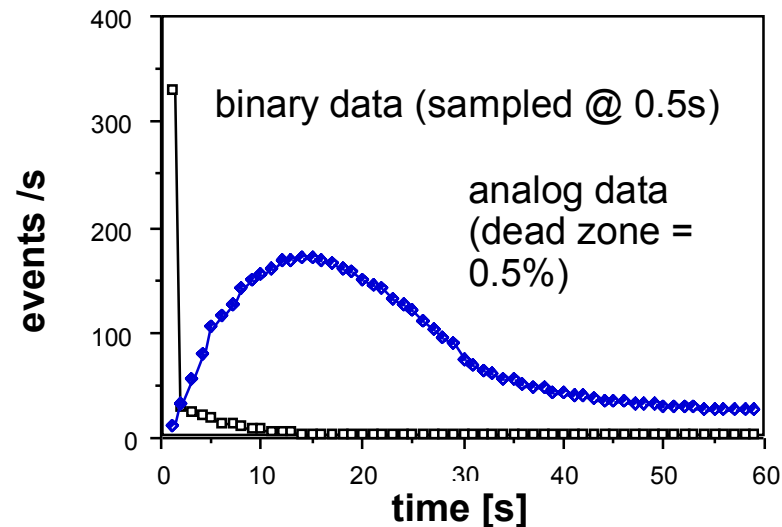
Since events are spread evenly over the DDS, no queue builds up as long as the event rate does not pass 286 per second

Worst case situation: loss of secondary power.

2500 binary events occur in the first second, but few in the following seconds. With automatic reconnection, a second peak can occur.

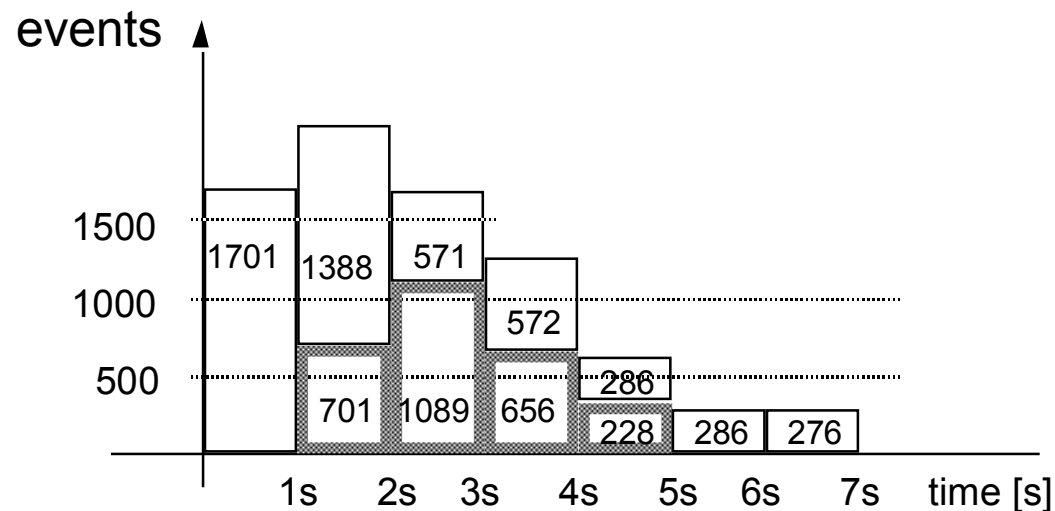
The analog avalanche causes about 100 changes in the first 2 seconds and 40 in the following 40 seconds:

binary and analog
avalanches:



Where is the bottleneck ?

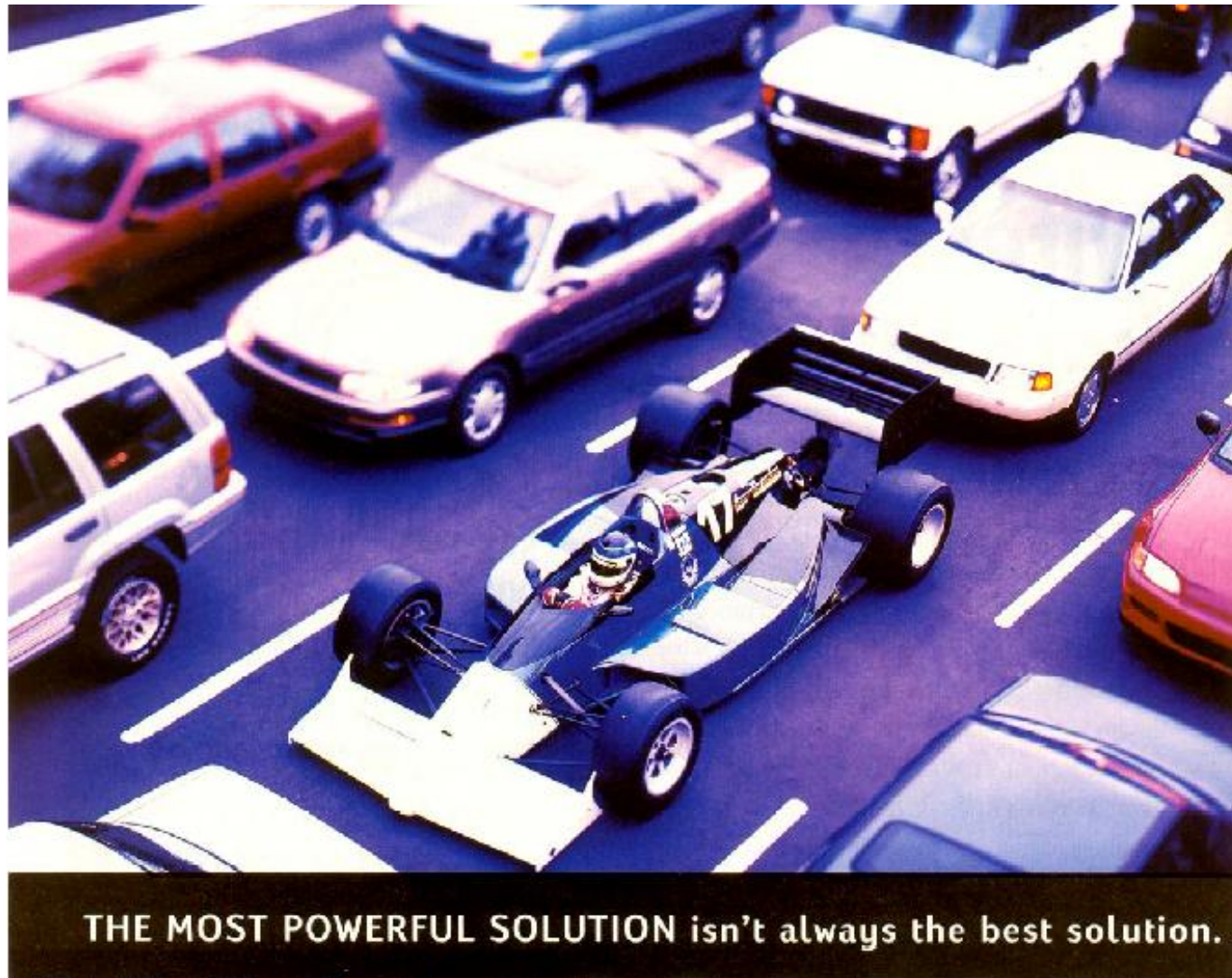
It can take up to 7 s until the avalanche is absorbed, i.e. until the operator has access to any particular variable.



Even in the worst case, the communication load over the Ethernet does not present a problem, since the production of events by the devices cannot exceed 1/15 ms, representing 0,33 % of the Ethernet's bandwidth.

The bottleneck was not the Ethernet capacity as was assumed, but the insufficient processing power of the operator workstations....

Always consider the whole system....



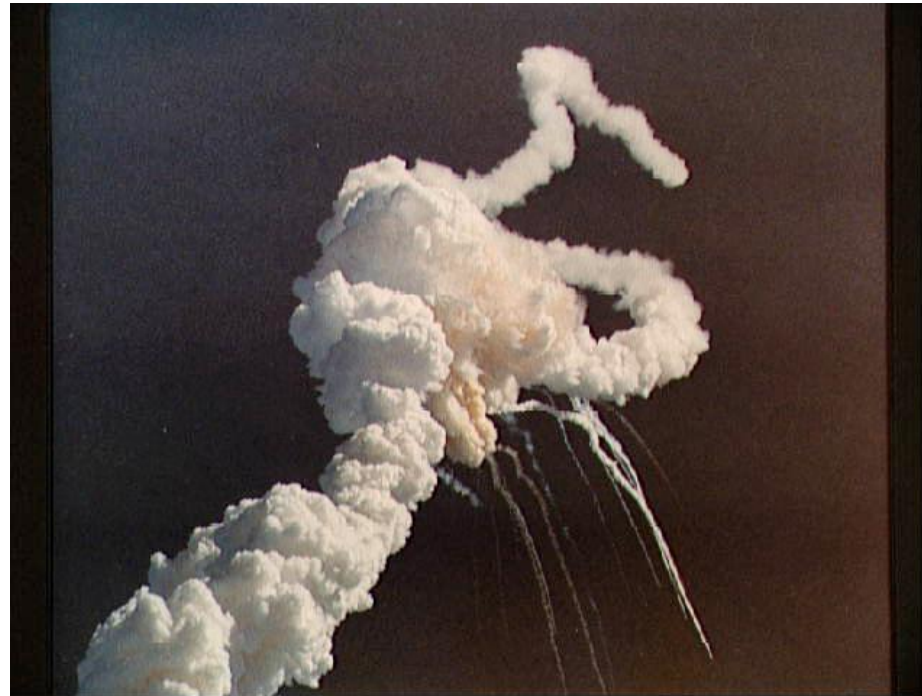
Conclusions

- Determinism is a basic property required of a critical control and protection system. A non-deterministic system is a "fair-weather" solution.
- A deterministic control system guarantees that all critical data are delivered within a fixed interval of time, or not at all.
- A deterministic system operates in normal time under worst-case conditions - this implies that resources seem wasted.
- The whole path from application to application (production, transmission and processing) must be deterministic, it is not sufficient that e.g. the medium access be deterministic.
- One can prove correctness of a deterministic system, but one cannot prove that a non-deterministic system is correct.
- Any non-deterministic delay in the path requires performance analysis to prove that it would work with a certain probability under realistic stress conditions.

Assessment

- 1 What is the difference between soft and hard real-time ?
- 2 What does determinism means and what does it allow to assess ?
- 3 What is to be done when non-deterministic components are present ?
- 4 What are the advantages and disadvantages of event-driven vs. cyclic systems ?
- 4 Can the response time of a hard real-time system be exactly predicted ?
- 5 Under which conditions can non-deterministic components be used ?





9.1 **Dependability - Overview**

Sûreté de fonctionnement - Vue d'ensemble

Verlässlichkeit - Übersicht

Prof. Dr. H. Kirrmann & Dr. B. Eschermann
ABB Research Center, Baden, Switzerland

Control Systems Dependability

9.1: Overview Dependable Systems

- Definitions: Reliability, Safety, Availability etc.,
- Failure modes in computers

9.2: Dependability Analysis

- Combinatorial analysis
- Markov models

9.3: Dependable Communication

- Error detection: Coding and Time Stamping
- Persistency

9.4: Dependable Architectures

- Fault detection
- Redundant Hardware, Recovery

9.5: Dependable Software

- Fault Detection,
- Recovery Blocks, Diversity

9.6: Safety analysis

- Qualitative Evaluation (FMEA, FTA)
- Examples

Motivation for Dependable Systems

Systems - if not working properly in a particular situation - may cause

- large losses of property or money
- injuries or deaths of people

To avoid such effects, these “mission-critical” systems must be designed specially so as to achieve a given behaviour in case of failure.

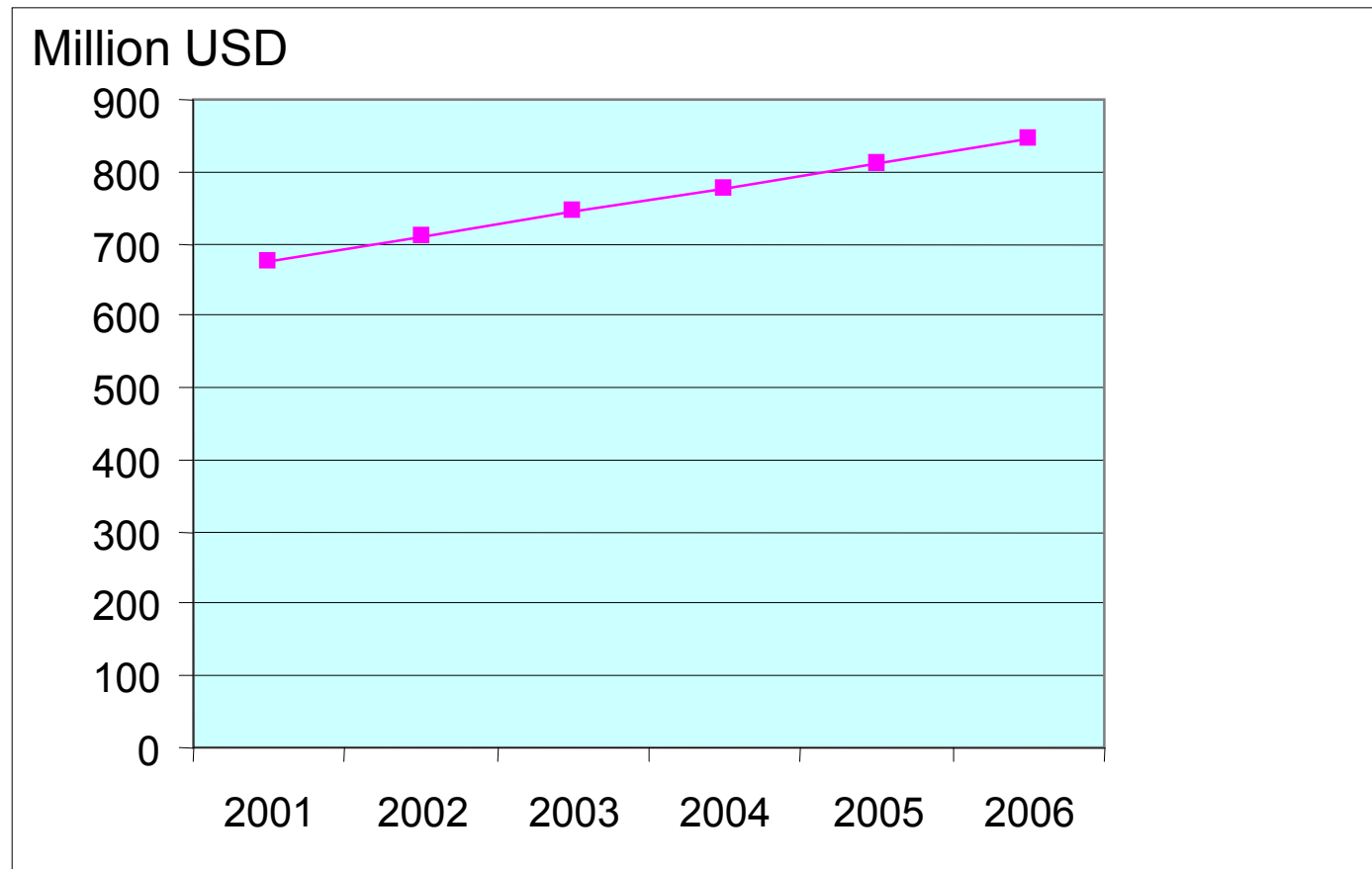
The necessary precautions depend on

- the probability that the system is not working properly
- the consequences of a system failure
- the risk of occurrence of a dangerous situation
- the negative impact of an accident (severity of damage, money lost)

Application areas for dependable systems

Space Applications	Launch rockets, Shuttle, Satellites, Space probes
Transportation	Airplanes (fly-by-wire), Railway signalling, Traffic control, Cars (ABS, ESP, brake-by-wire, steer-by-wire)
Nuclear Applications	Nuclear power plants, Nuclear weapons, Atomic-powered ships and submarines
Networks	Telecommunication networks, Power transmission networks, Pipelines
Business	Electronic stock exchange, Electronic banking, Data stores for Indispensable business data
Medicine	Irradiation equipment, Life support equipment
Industrial Processes	Critical chemical reactions, Drugs, Food

Market for safety- and critical control systems



increases more rapidly than the rest of the automation market

source: ARC Advisory group, 2002, Asish Ghosh

Definitions: Failure, Fault

A *mission* is the intended (specified) function of a device.

A *failure* (Ausfall, défaillance) is the non-fulfilment of this mission.

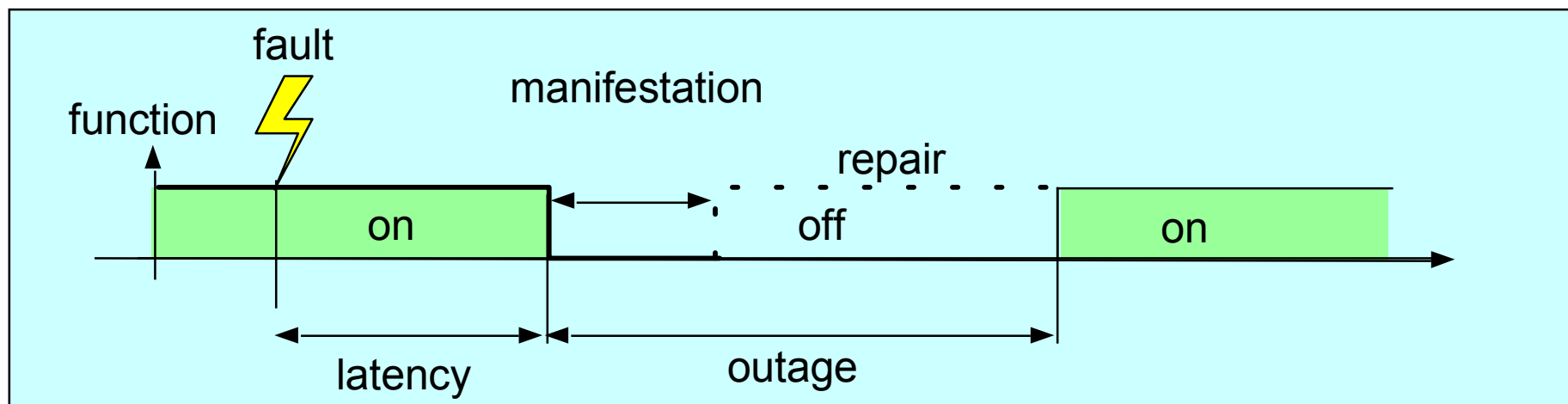
("termination of the ability of an item to perform its required function").

failures may be:

- momentary = outage (*Aussetzen*, raté)
- temporary = need repair = breakdown (*Panne*, panne) - for repairable systems only -
- definitive = (*Misserfolg*, échec)

A *fault* (Fehler, défaut) is the cause of a failure, it may occur long before the failure.

These terms can be applied to the whole system, or to elements thereof.



Fault, Error, Failure

Fault: missing or wrong functionality

- permanent: due to irreversible change, consistent wrong functionality (e.g. short circuit between 2 lines)
- intermittent: sometimes wrong functionality, recurring (e.g. loose contact)
- transient: due to environment, reversible if environment changes (e.g. electromagnetic interference)

Error: logical manifestation of a fault in an application

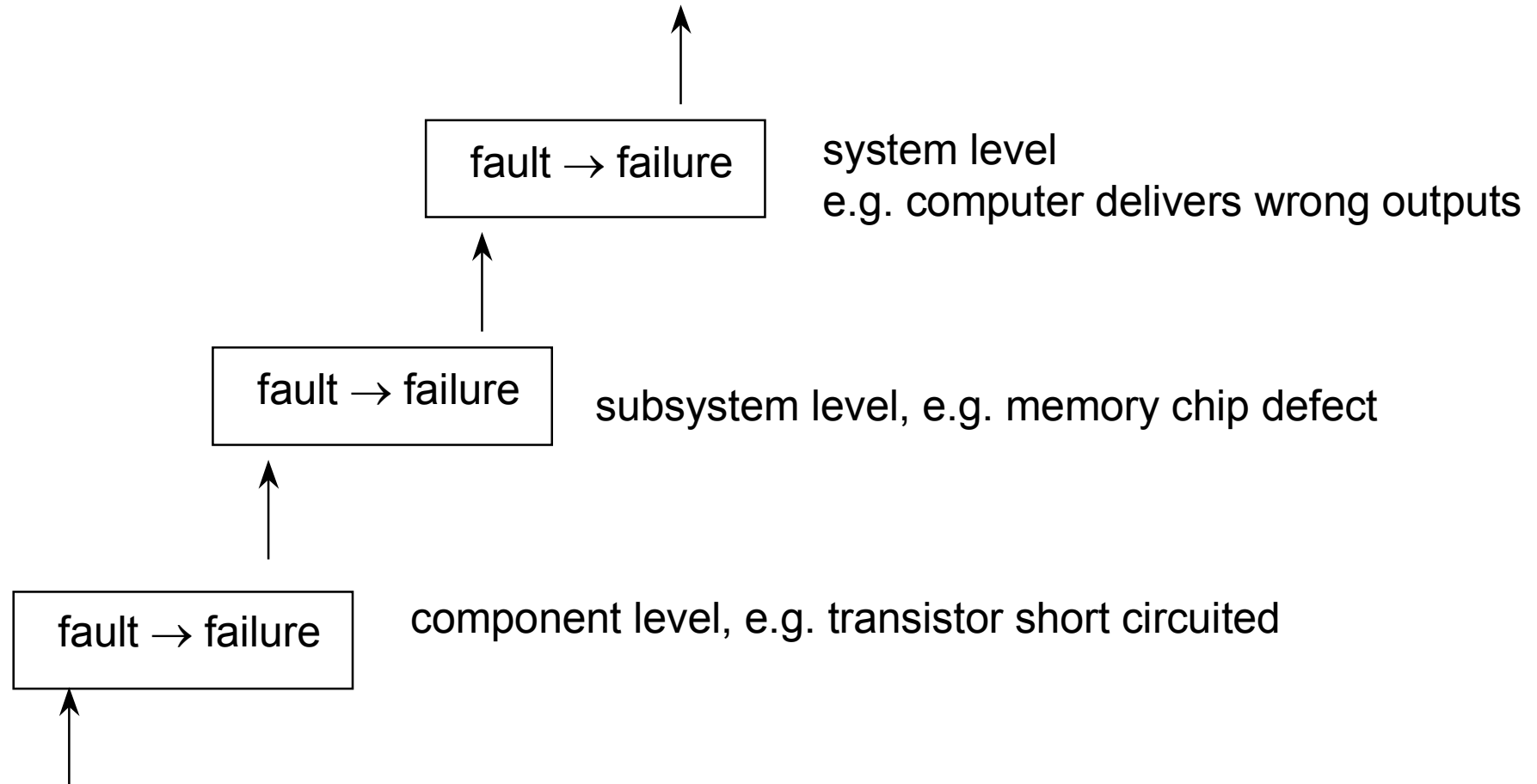
(e.g. short circuit leads to computation error if 2 lines carry different signals)

Failure: to perform a prescribed function

(e.g. if different signals on both lines lead to wrong output of chip)

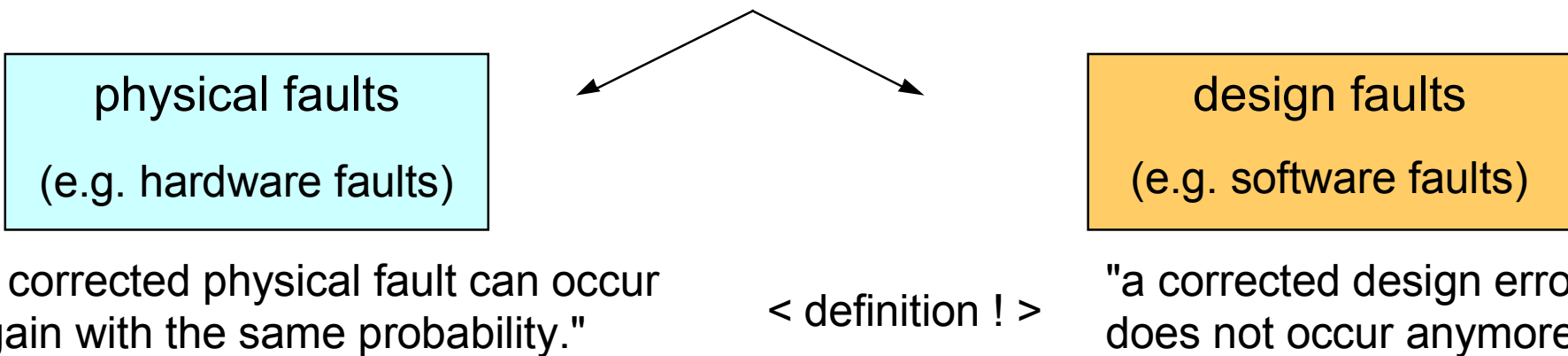


Hierarchy of Faults/Failures



Types of Faults

Computers can be affected by two kinds of faults:



Faults are originated by other faults (causality chain).

Physical faults can originate in design faults (e.g. missing cooling fan)

Most work in fault-tolerant computing addresses the physical faults, because it is easy to provide redundancy for the hardware elements.

Redundancy of the design means that several designs are available.

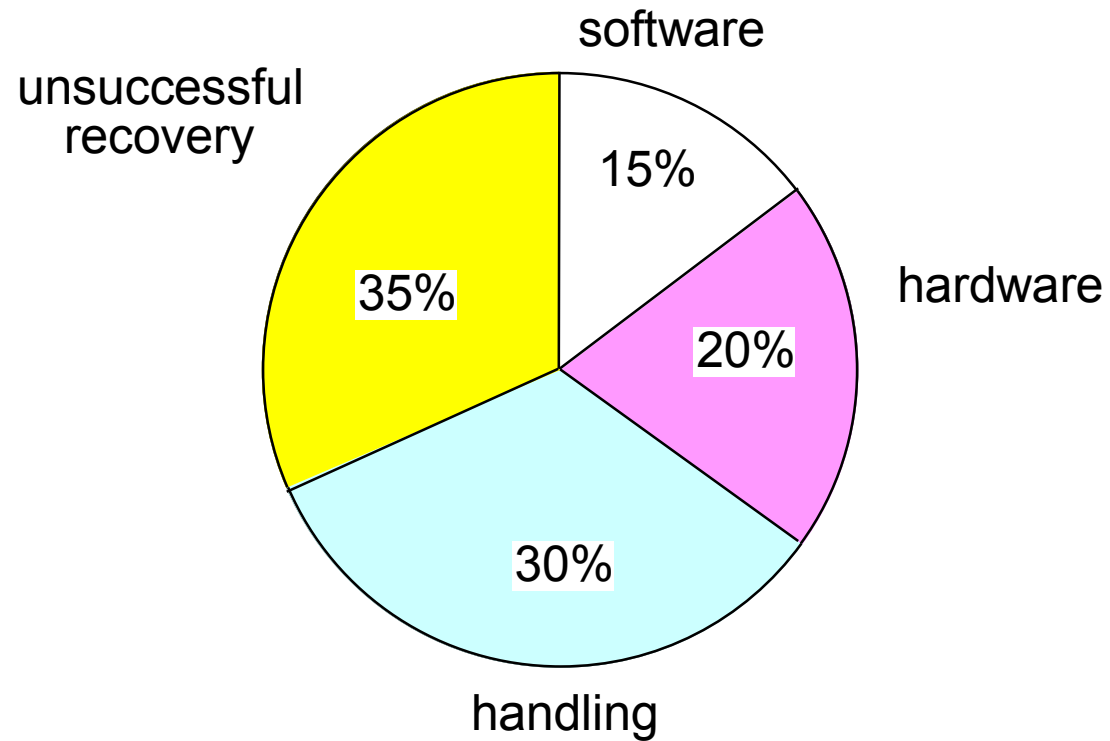
Random and Systematic Errors

Systematic errors are reproducible under given input conditions
Random Error appear with no visible pattern.

Although random errors are often associated with hardware errors and systematic errors with software errors, this needs not be the case

Transient errors , firm errors, soft errors,.... do not use these terms

Example: Sources of Failures in a telephone exchange



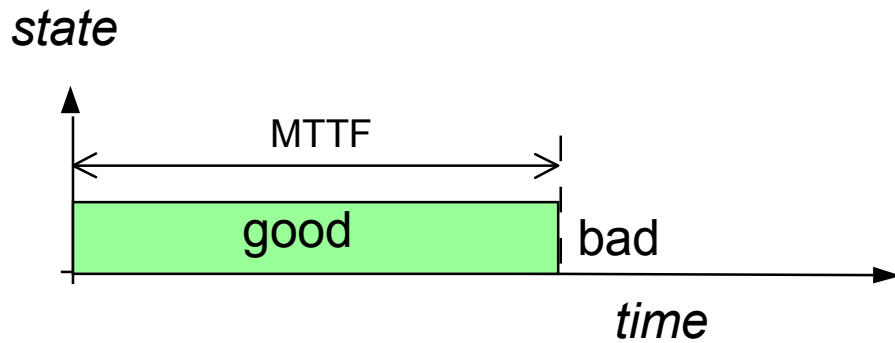
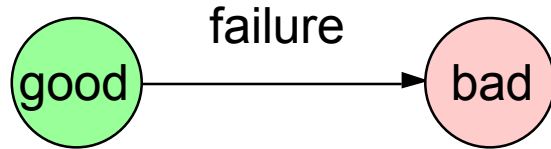
source: Troy, ESS1 (Bell USA)

Basic concepts

Basic concepts

Reliability and Availability

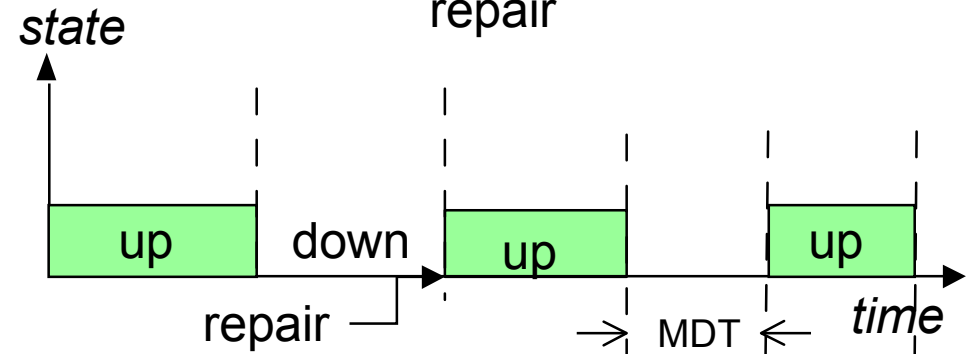
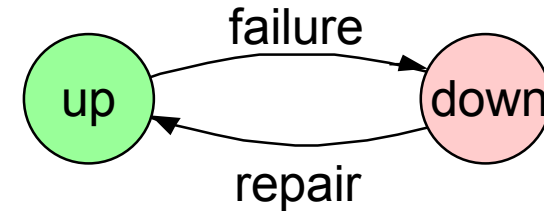
Reliability



definition: "probability that an item will perform its required function in the specified manner and under specified or assumed conditions *over a given time period*"

expressed shortly by its
MTTF: Mean Time To Fail

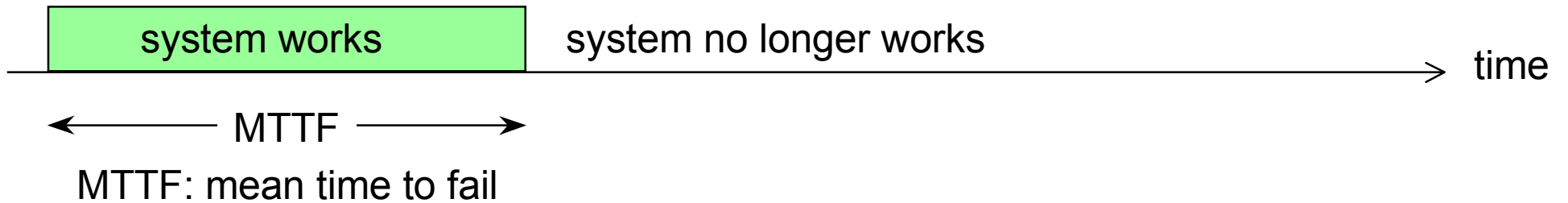
Availability



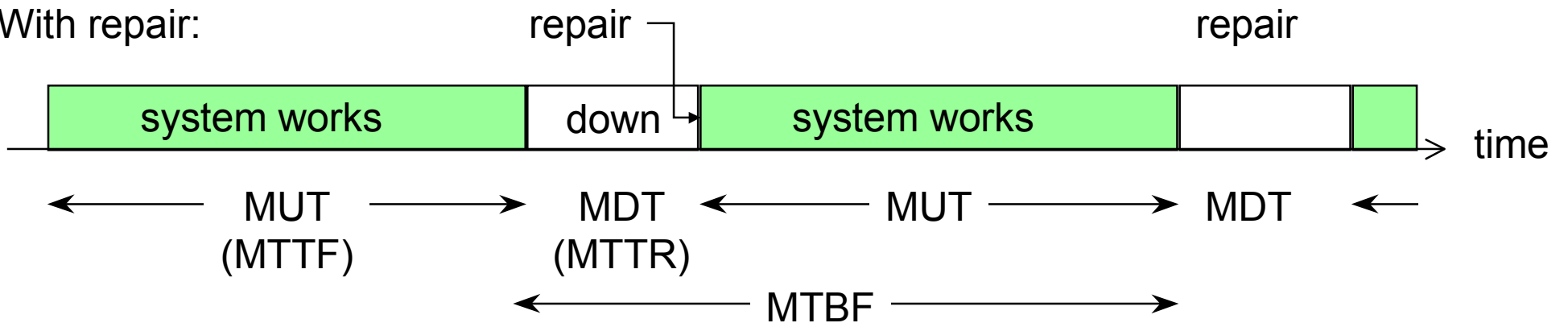
definition: "probability that an item will perform its required function in the specified manner and under specified or assumed conditions *at a given time*"

Failure/Repair Cycle

Without repair:



With repair:



MTTR: mean time to repair ~ MDT (mean down time)

MTBF: mean time between failures, (*n'est pas "moyenne des temps de bon fonctionnement")

$MTBF = MTTF + MTTR$

if $MTTR \ll MTTF$: $MTBF \approx MTTF$

Redundancy

Increasing safety or availability requires the introduction of redundancy (resources which are not needed if there were no failures).

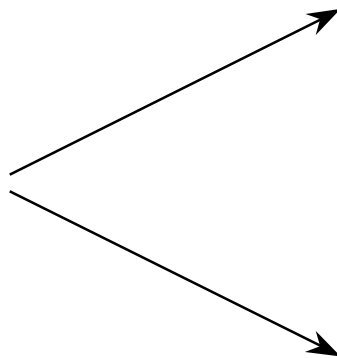
Faults are detected by introducing a **check redundancy**.

Operation is continued thanks to **operational redundancy** (can do the same task)

Increasing reliability and maintenance quality increases both safety and availability



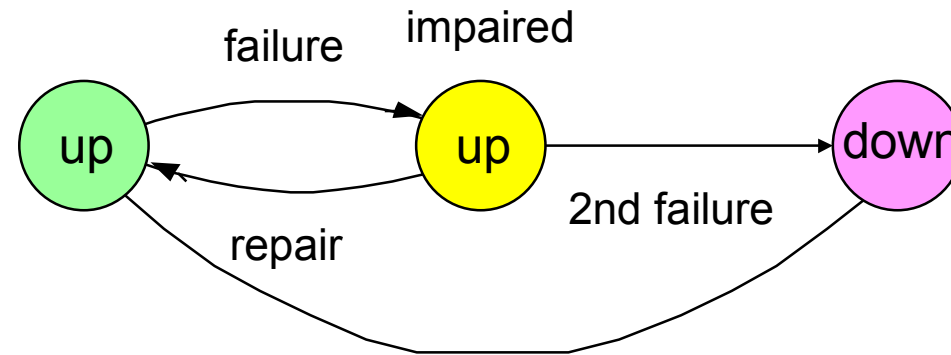
detected
fault
(don't know
about failure)



switch to red:
no accident risk (safe)
decreased traffic performance

switch to green:
accident risk
traffic continues (available)

Availability and Repair in redundant systems



unsuccessful switchover or common mode of failure

When redundancy is available, the system does not fail until redundancy is exhausted (or redundancy switchover is unsuccessful)

Maintenance

"The combination of all technical and administrative actions, including supervision actions intended to retain a component in, or restore it to, a state in which it can perform its required function"

Maintenance takes the form of

- **corrective maintenance**: executed when a part actually fails (repair)

"go to the garage when the motor fails"

- **preventive maintenance**: restoring redundancy

and in particular restore degraded parts to error-free state

"go to the garage to change oil and pump up the reserve tyre"

- **scheduled maintenance** (time-based maintenance)

"go to the garage every year"

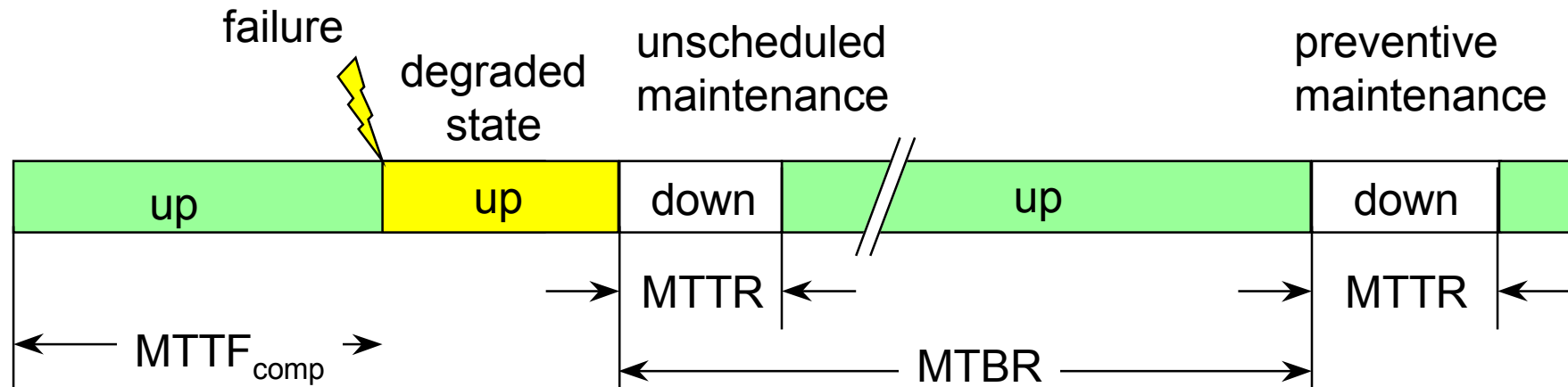
- **predictive maintenance** (condition-based maintenance)

"go to the garage at the next opportunity since motor heats up"

preventive maintenance does not necessarily stop production if redundancy is available

"differed maintenance" is performed in a non-productive time.

Differed maintenance



Redundancy does not replace maintenance:
it allows to differ maintenance to a convenient moment
(e.g. between 02h00 and 04h00 in the morning).

The system may remain on-line or be taken shortly out of operation.

The mean time between repairs (MTBR) expressed how often any component fails

The mean time between failure concerns the whole system.

Differed maintenance is only interesting for plants that are not fully operational 24/24.

Preventive maintenance

In principle, preventive maintenance restores the initially good state at regular intervals.

This assumes that the coverage of the tests is 100% and that no uncorrected aging takes place.

Safety

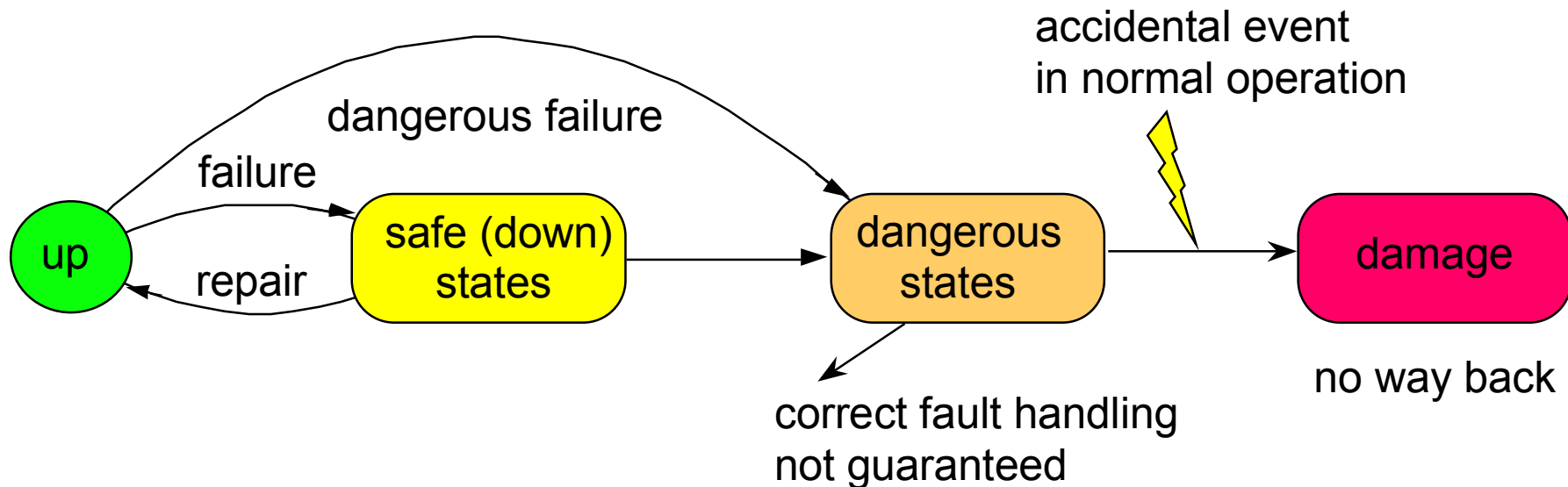
we distinguish:

- hazards caused by the presence of control system itself:
explosion-proof design of measurement and control equipment
(e.g. Ex-proof devices, see "Instrumentation")
- implementation of safety regulation (protection) by control systems
"safety"- PLC, "safety" switches
(requires tamper-proof design)
protection systems in the large
(e.g. Stamping Press Control (*Pressesteuerungen*),
Burner Control (*Feuerungssteuerungen*))
- hazard directly caused by malfunction of the control system
(e.g. flight control)

Safety

The probability that the system does not behave in a way considered as dangerous.

Expressed by the probability that the system does not enter a state defined as dangerous



difficulty of defining which states are dangerous -
level of damage ? acceptable risk ?

Safe States

Safe state

- exists: sensitive system
- does not exist: critical system

Sensitive systems

- railway: train stops, all signals red (**but**: fire in tunnel?)
- nuclear power station: switch off chain reaction by removing moderator (may depend on how reactor is constructed)

Critical systems

- military airplanes: only possible to fly with computer control system (plane inherently unstable)

Types of Redundancy

Structural redundancy (hardware):

Extend system with additional components that are not necessary to achieve the required functionality (e.g. overdimension wire gauge, use 2-out-of-3 computers)

Functional redundancy (software):

Extend the system with unnecessary functions

- additional functions (e.g. for error detection or to switch to standby unit)
- diversity (additional different implementation of the required functions)

Information redundancy:

Encode data with more bits than necessary
(e.g. parity bit, CRC, 1-out-of-n-code)

Time redundancy:

Use additional time, e.g. to do checks or to repeat computation

Availability and Safety (1)

Availability

availability is an economical objective.

high availability increases
production time and yield
(e.g. airplanes are aloft)

The gain can be measured in
additional up-time

availability depends on a
functional redundancy (which can
take over the function) and on the
quality of maintenance

Safety

safety is a regulatory objective

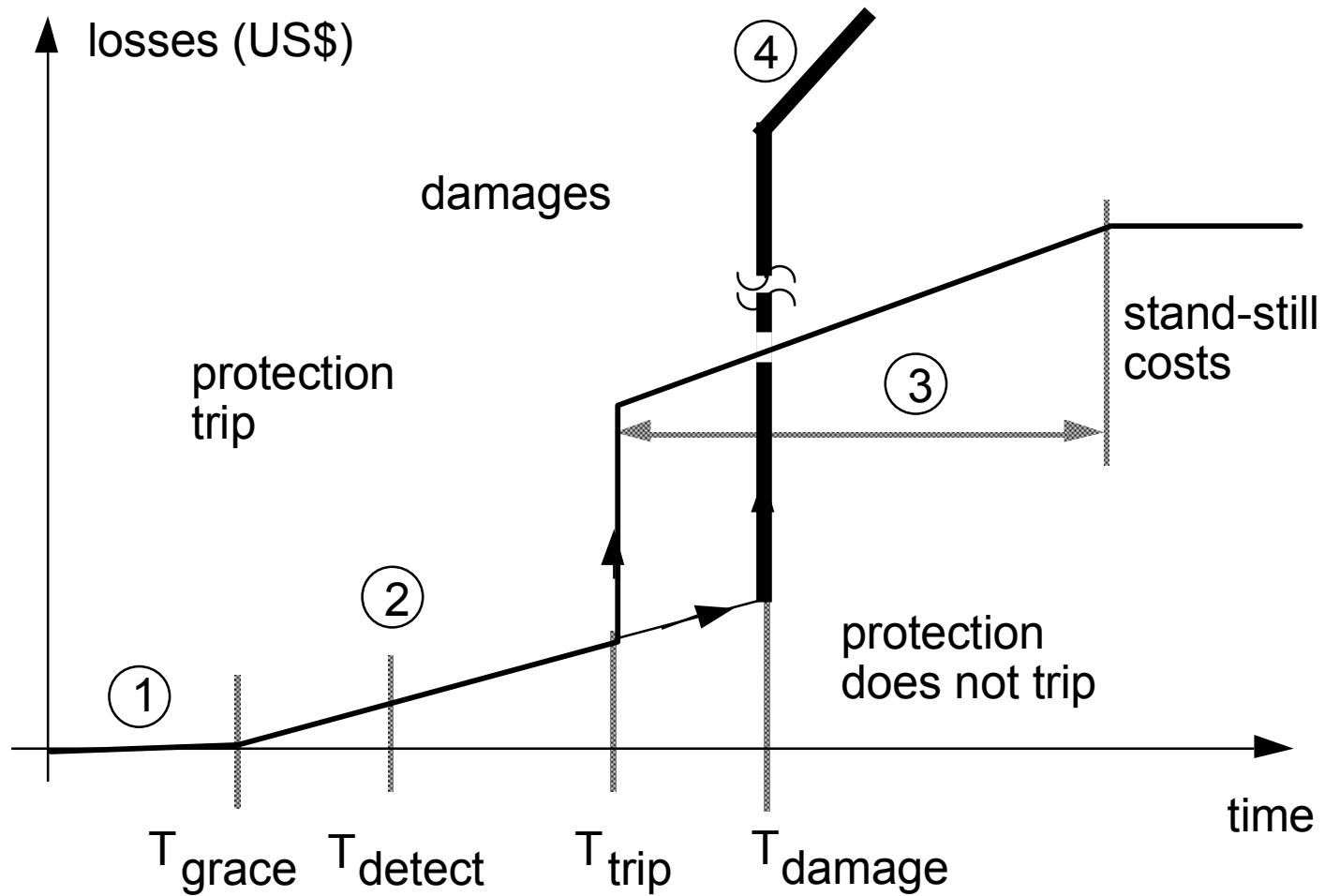
high safety reduces the
risk to the process and its
environment

The gain can be measured in
lower insurance rates

safety depends on the introduction of
check redundancy (fail-stop
systems) and/or functional
redundancy (fail-operate systems)

Safety and Availability are often contradictory (completely safe systems are unavailable) since they share redundancy.

Cost of failure in function of duration



Safety and Security

Safety (*Sécurité, Sicherheit*):

Avoid dangerous situations due to unintentional failures

–failures due to random/physical faults

–failures due to systematic/design faults

e.g. railway accident due to burnt out red signal lamp

e.g. rocket explosion due to untested software (→ Ariane 5)

Security (*Sécurité informatique, IT-Sicherheit*):

Avoid dangerous situations due to malicious threats

–authenticity / integrity (*intégrité*): protection against tampering and forging

–privacy / secrecy (*confidentialité, Vertraulichkeit*): protection against eavesdropping

e.g. robbing of money tellers by using weakness in software

e.g. competitors reading production data

The boundary is fuzzy since some unintentional faults can behave maliciously.

(*Sûreté: terme général: aussi probabilité de bon fonctionnement, Verlässlichkeit*)

How to Increase Dependability?

Fault tolerance: Overcome faults without human intervention.

Requires **redundancy**: Resources normally not needed to perform the required function.

Check Redundancy (that can detect incorrect work)

Functional Redundancy (that can do the work)

Contradiction: Fault-tolerance increases complexity and failure rate of the system.

Fault-tolerance is no panacea: Improvements in dependability are in the range of 10..100.

Fault-tolerance is costly:

x 3 for a safe system,

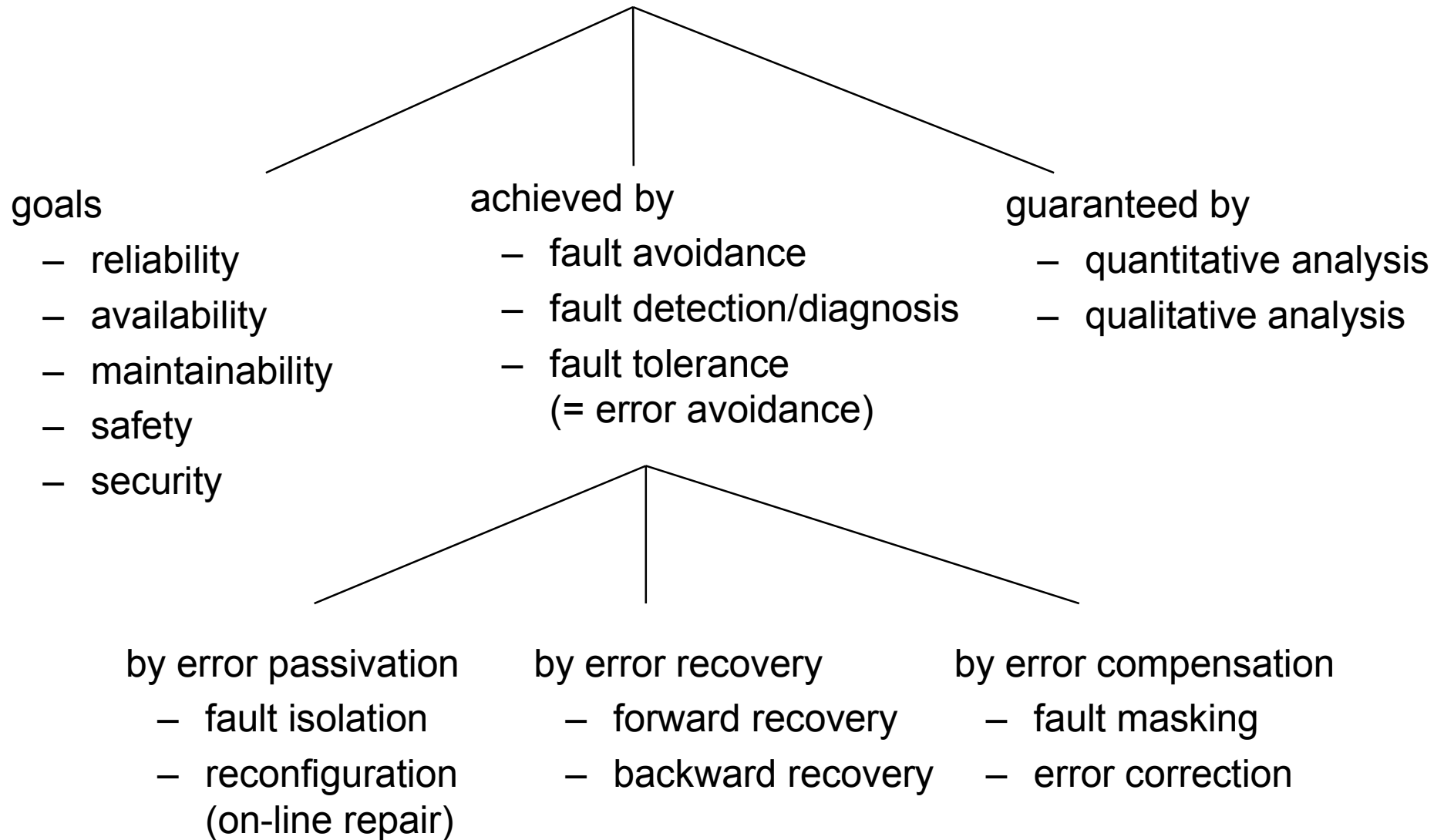
x 4 times for an available 1oo2 system (1-out-of-2),

x 6 times for a 2oo3 (2-out-of-3) voting system

Fault-tolerance is no substitute for quality

Dependability

(*Sûreté de fonctionnement, Verlässlichkeit*)



Failure modes in computers

9.1: Overview Dependable Systems

- Definitions: Reliability, Safety, Availability etc.,
- **Failure modes in computers**

9.2: Dependability Analysis

- Combinatorial analysis
- Markov models

9.3: Dependable Communication

- Error detection: Coding and Time Stamping
- Persistency

9.4: Dependable Architectures

- Fault detection
- Redundant Hardware, Recovery

9.5: Dependable Software

- Fault Detection,
- Recovery Blocks, Diversity

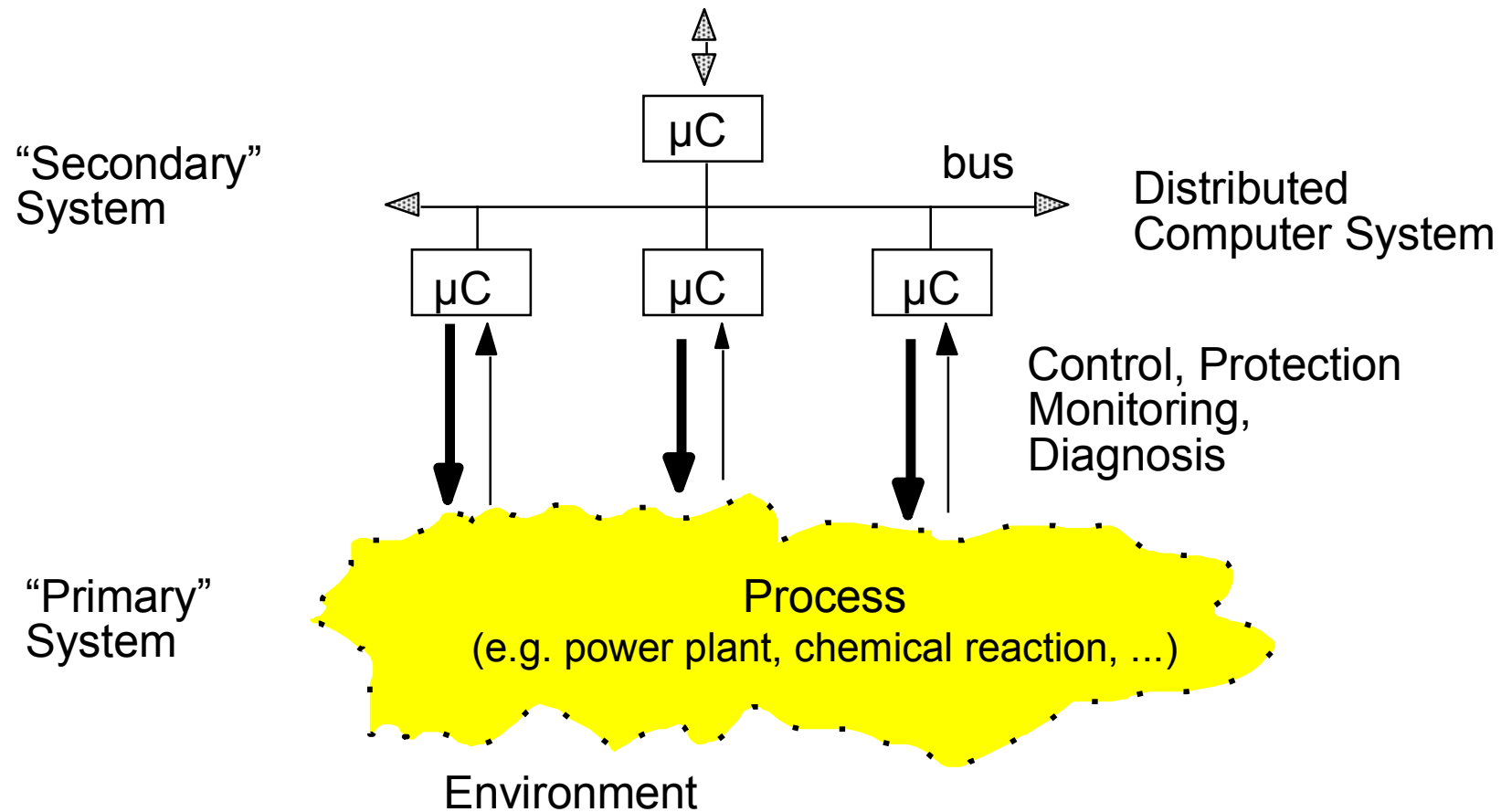
9.6: Safety analysis

- Qualitative Evaluation (FMEA, FTA)
- Examples

Failure modes in computers

Safety or availability can only be evaluated considering the total system controller + plant.

Computers and Processes



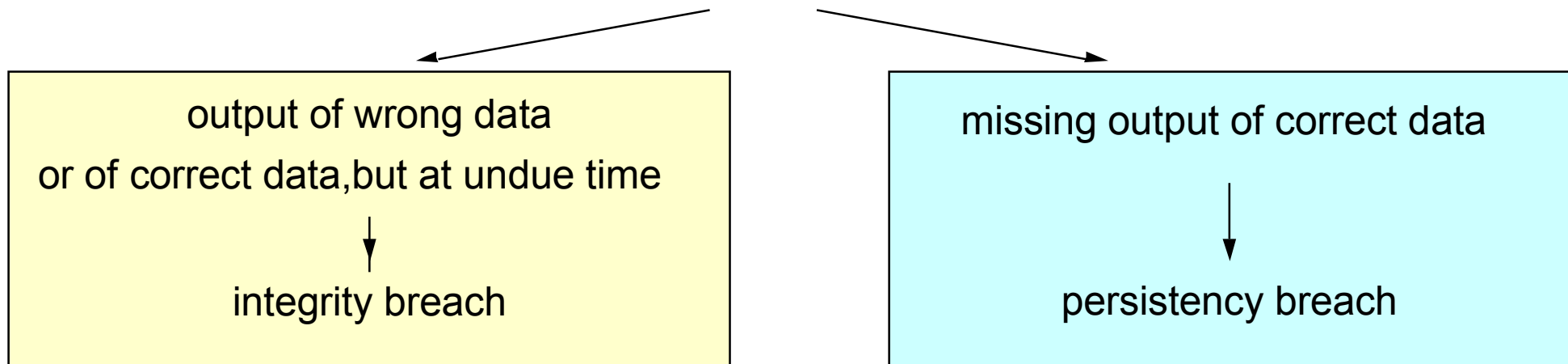
Availability/safety depends on **output** of computer system and process/environment.

Types of Computer Failures

Computers can fail in a number of ways

Breach of the specifications = does not behave as intended

reduced to two cases



Fault-tolerant computers allow to overcome these situations.

The architecture of the fault-tolerant computer depends on the encompassed dependability goals

Safety Threats

depending on the controlled process,
safety can be threatened by failures of the control system:

integrity breach

not recognized, wrong data, or correct data, but at the wrong time

if the process is irreversible

(e.g. closing a high power breaker,
banking transaction)

Requirement:

fail-silent (fail-safe, fail-stop) computer
"rather stop than fail"

persistency breach

no usable data, loss of control

if the process has no safe side

(e.g. landing aircraft)

Requirement:

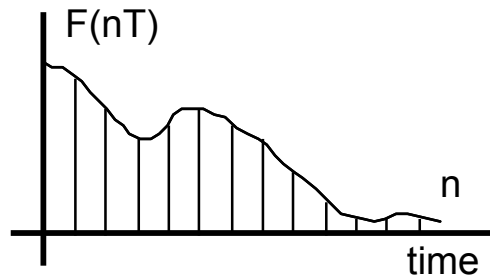
fail-operate computer
"rather some wrong data than none"

Safety depends on the tolerance of the process against failure of the control system

Plant type and dependability

continuous systems

modelled by differential equations, and in the linear case, by Laplace or z-transform (sampled)



continuous systems are generally reversible.

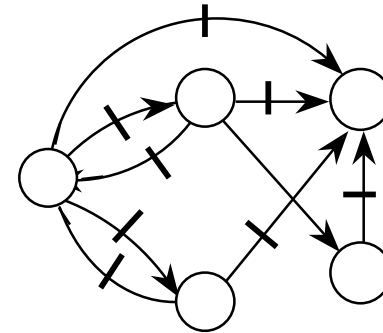
tolerates sporadic, wrong inputs during a limited time (similar: noise)

tolerate loss of control only during a short time.

require persistent control

discrete systems

modelled by state machines, Petri nets, Grafcet,....



transitions between states are normally irreversible.

do not tolerate wrong input.
difficult recovery procedure

tolerate loss of control during a relatively long time (remaining in the same state is in general safe).

require integer control

Persistency/Integrity by Application Examples

secondary system	primary system	availability	safety
integrity		substation protection	railway signalling
persistency		airplane control	

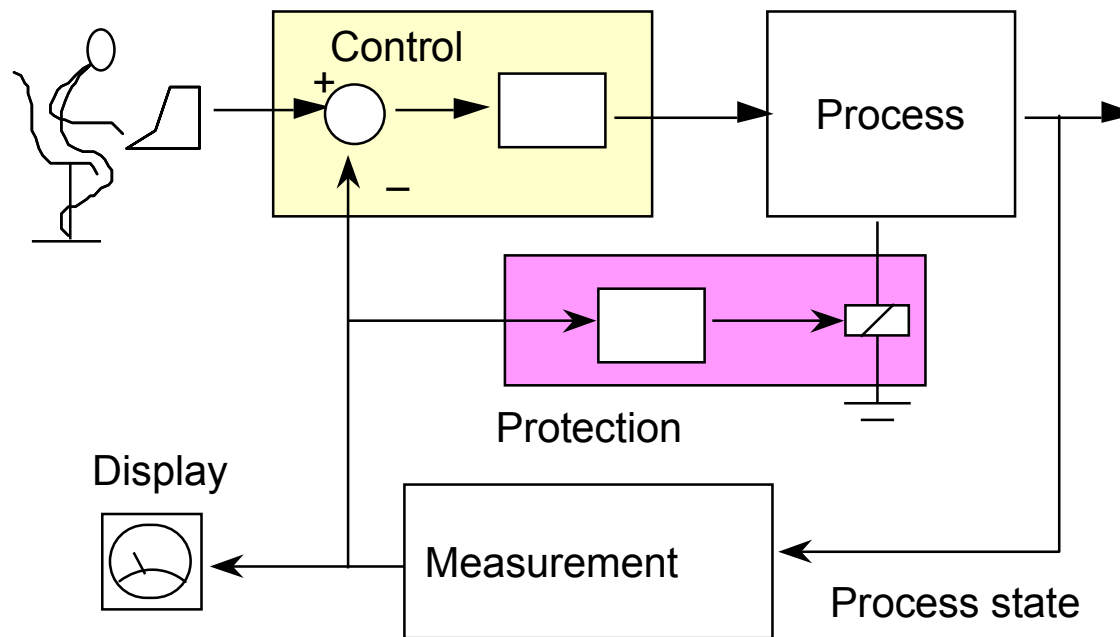
Protection and Control Systems

Control system:

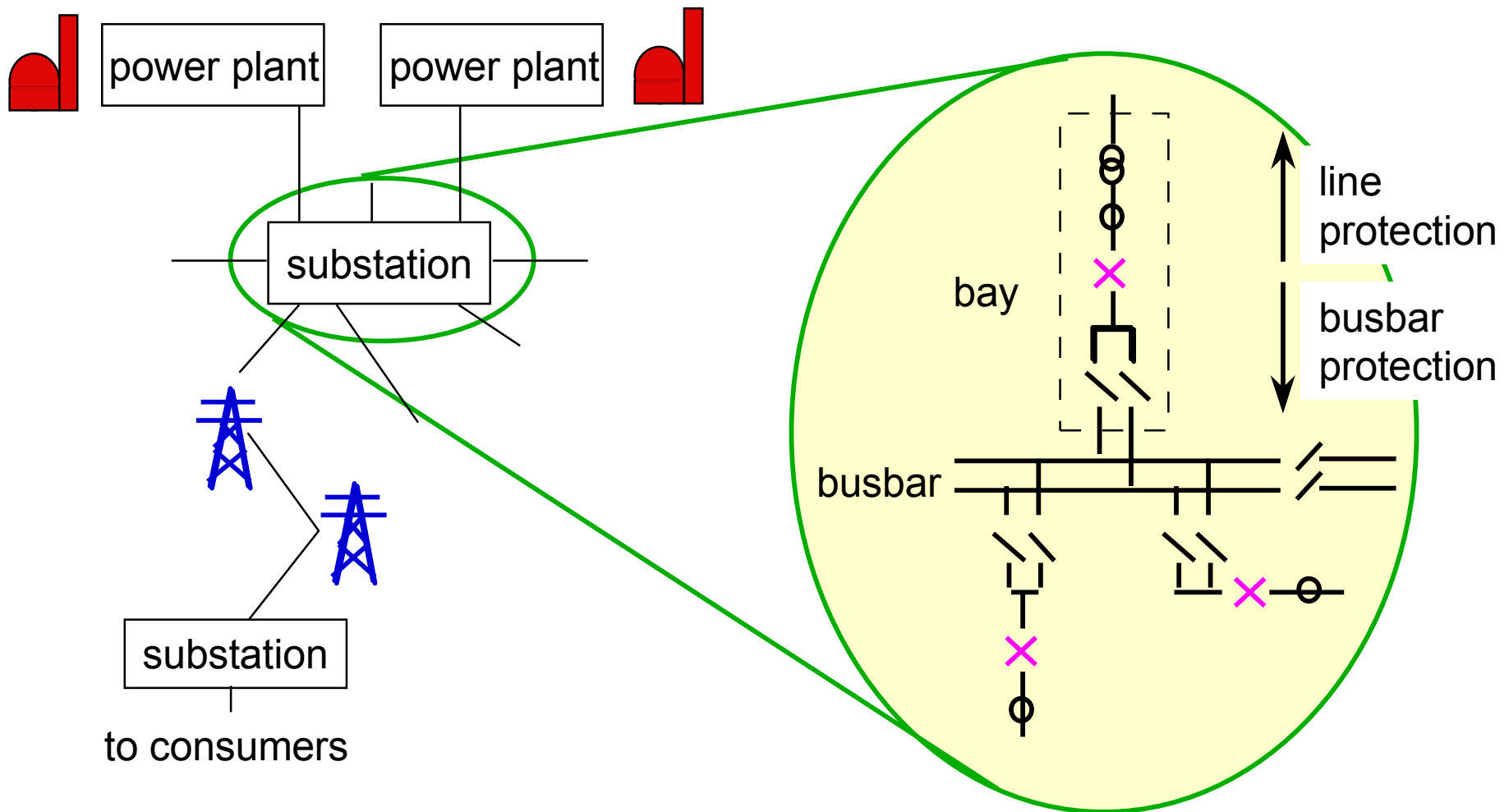
Continuous non-stop operation
(open or closed loop control)
Maximal failure rate given in
failures per hour.

Protection system:

Not acting normally,
forces safe state (trip) if necessary
Maximal failure rate given in failures per
demand.



Example Protection Systems: High-Voltage Transmission



Two kinds of malfunctions:

An underfunction (not working when it should) of a protection system is a safety threat

An overfunction (working when it should not) of a protection system is an availability threat

Findings

Reliability and fault tolerance must be considered early in the development process, they can hardly be increased afterwards.

Reliability is closely related to the concept of quality, its roots are laid in the design process, starting with the requirement specs, and accompanying through all its lifetime.

References

H. Nussbaumer: Informatique industrielle IV; PPUR.

J.-C. Laprie (ed.): Dependable computing and fault tolerant systems; Springer.

J.-C. Laprie (ed.): Guide de la sûreté de fonctionnement; Cépaduès.

D. Siewiorek, R. Swarz: The theory and practice of reliable system design; Digital Press.

T. Anderson, P. Lee: Fault tolerance - Principles and practice; Prentice-Hall.

A. Birolini: Quality and reliability of technical systems; Springer.

M. Lyu (ed.): Software fault tolerance; Wiley.

Journals: IEEE Transactions on Reliability, IEEE Transactions on Computers

Conferences: International Conference on Dependable Systems and Networks,
European Dependable Computing Conference

Assessment

which kinds of fault exist and how are they distinguished

explain the difference between reliability, availability, safety in terms of a state diagram

explain the trade-off between availability and safety

what is the difference between safety and security

explain the terms MTTF, MTTR, MTBF, MTBR

how does a protection system differ from a control system when considering failures ?

which forms of redundancy exist for computers ?

how does the type of plant influence its behaviour towards faults ?