AMERICAN NATIONAL STANDARD

ANSI/ISA–88.00.01–2010

Batch Control Part 1: Models and Terminology

Approved 6 December 2010

Get more FREE standards from Standard Sharing Group and our chats

ANSI/ISA-88.00.01-2010
Batch Control Part 1: Models and Terminology

ISBN: 978-1-936007-75-2

Copyright © 2010 by ISA. All rights reserved. Not for resale. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), without the prior written permission of the Publisher.

ISA
67Alexander Drive
P. O. Box 12277
Research Triangle Park, North Carolina 27709
USA

CONTENTS

Get more FREE standards from Standard Sharing Group and our chats

Get more FREE standards from Standard Sharing Group and our chats

Figures

Tables

Get more FREE standards from Standard Sharing Group and our chats

Batch Control Part 1: Models and terminology

FOREWORD

This foreword, as well as all footnotes and annexes, is included for information purposes and is not part of ANSI/ISA-88.00.01-2010.

This document has been prepared as part of the service of ISA, The International Society of Automation, toward a goal of uniformity in the field of automation. To be of real value, this document should not be static but should be subject to periodic review. Toward this end, the Society welcomes all comments and criticisms and asks that they be addressed to the Secretary, Standards and Practices Board; ISA; 67 Alexander Drive; P. O. Box 12277; Research Triangle Park, NC 27709; Telephone (919) 549-8411; Fax (919) 549-8288; E-mail: standards@isa.org.

The ISA Standards and Practices Department/ endeavours to introduce SI-acceptable metric units in all new and revised standards, recommended practices, and technical reports to the greatest extent possible. Standard for Use of the International System of Units (SI): The Modern Metric System, published by the American Society for Testing & Materials as IEEE/ASTM SI 10-97, and future revisions, will be the reference guide for definitions, symbols, abbreviations, and conversion factors.

It is the policy of ISA to encourage and welcome the participation of all concerned individuals and interests in the development of ISA standards, recommended practices, and technical reports. Participation in the ISA standards-making process by an individual in no way constitutes endorsement by the employer of that individual, of ISA, or of any of the standards, recommended practices, and technical reports that ISA develops.

CAUTION — ISA DOES NOT TAKE ANY POSITION WITH RESPECT TO THE EXISTENCE OR VALIDITY OF ANY PATENT RIGHTS ASSERTED IN CONNECTION WITH THIS DOCUMENT, AND ISA DISCLAIMS LIABILITY FOR THE INFRINGEMENT OF ANY PATENT RESULTING FROM THE USE OF THIS DOCUMENT. USERS ARE ADVISED THAT DETERMINATION OF THE VALIDITY OF ANY PATENT RIGHTS, AND THE RISK OF INFRINGEMENT OF SUCH RIGHTS, IS ENTIRELY THEIR OWN RESPONSIBILITY.

PURSUANT TO ISA'S PATENT POLICY, ONE OR MORE PATENT HOLDERS OR PATENT APPLICANTS MAY HAVE DISCLOSED PATENTS THAT COULD BE INFRINGED BY USE OF THIS DOCUMENT AND EXECUTED A LETTER OF ASSURANCE COMMITTING TO THE GRANTING OF A LICENSE ON A WORLDWIDE, NON-DISCRIMINATORY BASIS, WITH A FAIR AND REASONABLE ROYALTY RATE AND FAIR AND REASONABLE TERMS AND CONDITIONS. FOR MORE INFORMATION ON SUCH DISCLOSURES AND LETTERS OF ASSURANCE, CONTACT ISA OR VISIT WWW.ISA.ORG/STANDARDSPATENTS.

OTHER PATENTS OR PATENT CLAIMS MAY EXIST FOR WHICH A DISCLOSURE OR LETTER OF ASSURANCE HAS NOT BEEN RECEIVED. ISA IS NOT RESPONSIBLE FOR IDENTIFYING PATENTS OR PATENT APPLICATIONS FOR WHICH A LICENSE MAY BE REQUIRED, FOR CONDUCTING INQUIRIES INTO THE LEGAL VALIDITY OR SCOPE OF PATENTS, OR DETERMINING WHETHER ANY LICENSING TERMS OR CONDITIONS PROVIDED IN CONNECTION WITH SUBMISSION OF A LETTER OF ASSURANCE, IF ANY, OR IN ANY LICENSING AGREEMENTS ARE REASONABLE OR NON-DISCRIMINATORY.

ISA REQUESTS THAT ANYONE REVIEWING THIS DOCUMENT WHO IS AWARE OF ANY PATENTS THAT MAY IMPACT IMPLEMENTATION OF THE DOCUMENT NOTIFY THE ISA STANDARDS AND PRACTICES DEPARTMENT OF THE PATENT AND ITS OWNER.

ADDITIONALLY, THE USE OF THIS DOCUMENT MAY INVOLVE HAZARDOUS MATERIALS, OPERATIONS OR EQUIPMENT. THE DOCUMENT CANNOT ANTICIPATE ALL POSSIBLE APPLICATIONS OR ADDRESS ALL POSSIBLE SAFETY ISSUES ASSOCIATED WITH USE IN HAZARDOUS CONDITIONS. THE USER OF THIS DOCUMENT MUST EXERCISE SOUND PROFESSIONAL JUDGMENT CONCERNING ITS USE AND APPLICABILITY UNDER THE USER'S PARTICULAR CIRCUMSTANCES. THE USER MUST ALSO CONSIDER THE APPLICABILITY OF ANY GOVERNMENTAL REGULATORY LIMITATIONS AND ESTABLISHED SAFETY AND HEALTH PRACTICES BEFORE IMPLEMENTING THIS DOCUMENT.

THE USER OF THIS DOCUMENT SHOULD BE AWARE THAT THIS DOCUMENT MAY BE IMPACTED BY ELECTRONIC SECURITY ISSUES. THE COMMITTEE HAS NOT YET ADDRESSED THE POTENTIAL ISSUES IN THIS VERSION.

This Part 1 standard is structured to follow IEC (International Electrotechnical Commission) guidelines.

This document is Part 1 of a multi-part set of standards that defines batch control. Other standards in the series include, under the general title "Batch Control":

- Part 2: Data structures and guidelines for languages
- Part 3: General and site recipe models and representation
- Part 4: Batch production records
- Part 5: Implementation models and terminology for modular equipment control (in development at the time of publication; visit www.isa.org/standards)

This revised Part 1 replaces ISA-88.01-1995. The major changes made to this standard from the previous version are:

Get more FREE standards from Standard Sharing Group and our chats

1. Models and text are modified to provide more detail and clarity. Key clarifications are:

    a. All recipe-aware equipment modules contain procedural control

    b. Execution of all procedural control contained directly in units is part of the unit supervision activity.

    c. The relationships between types of recipes, recipe components, and equipment control are more fully described and illustrated.

    d. Entity relationship diagrams have been replaced with more intuitive UML instance diagrams.

    e. The transition diagram for the procedural states example has been updated with a more intuitive and complete UML state diagram.

    f. References to other standards in the series and to the ANSI/ISA-95 and IEC 62264-1 are included to provide direction for further clarification of selected topics.

2. Previous Clauses 4 through 6 (now 4 through 8) were rearranged to provide a clearer top down organization of the document. Key changes are:

    a. Simultaneous characterization of the physical and equipment entity model levels at and below the process cell to eliminate redundancy and clarify that the control in these entities is the basis for partitioning this part of the physical model (see 4.3).

b. Combining the descriptions of basic, procedural, and coordination control with their usage in each type of equipment entity, providing a single consolidated discussion of each type of control (see Clause 5)

c. Additional considerations to support application of the models have been grouped in Clause 7 to clarify their supporting relationship to the core models.

3. Clause 9 was added to define completeness, compliance, and conformance in relation to this standard.

4. Annex B was added to more fully describe the changes in this document compared with the superseded 1995 version.

5. Annex C was added to clarify a number of points concerning the models, their application, and the new clause on conformance and compliance.

6. Annex D was added to provide a more expansive procedural state reference model. The model found in Clause 7 may be considered a collapsed version of this more general model.

The following served as active members of the ISA88 Part 1 Update Working Group:

Name	**Affiliation**
Paul Nowicki, WG Chairman/Editor	World Food Trace, Inc. / Heat and Control, Inc.
Randy Dwiggins, WG Secretary and ISA88 Chair	NNE Pharmaplan
Mark Albano	Honeywell
Shawn Baker	NNE Pharmaplan
Cindy Benedict	Siemens
Dennis Brandl	BR&L Consulting
Sean Brennan	Stone Technologies
Ed Bristol	Consultant
Dave Chappell	CMAa Consulting
Leo Charpentier	NovaTech
Lynn Craig	Consultant
Tom Crowl	Genentech
Em Delahostria	Rockwell Automation
Dave Emerson	Yokogawa
Herbert Falk	Sisco
Gus Finucci	EnteGreat
Alistair Gillanders	DynoChem Inc
Alex Habib	Consultant
Mike Hakes	Colt Engineering
Gavan Hood	Simul-Tech Pty Ltd
Baha Korkmaz	Invensys
Motoichi Kuwatani	Yokogawa
Francis Lovering	ControlDraw
Ed Lynch	Pfizer
Dawn Marruchella	Emerson Process Management
Thomas Müller-Heinzerling	Siemens
Steve Murray	Emerson Process Management
Thomas Nash	Areva
Albert Pampel	Consultant
Henry Salomons	Dow
Joseph Schipani	Invensys
Dan Seger	Rockwell Automation
Mike Smith	ABB
Marcus Tennent	Yokogawa
Jean Vieille	Control Chain Group
Frede Vinther	NNE Pharmaplan
Kate Waters	Genentech
Max Weinmann	Emerson Process Management

Get more FREE standards from Standard Sharing Group and our chats

INTRODUCTION

This Part 1 standard is structured to follow the IEC (International Electrotechnical Commission) guidelines. Therefore, the first three clauses discuss the Scope of the standard, Normative References, and Definitions, in that order.

The models and terminology in this standard are highly interdependent, making many of the definitions in Clause 3 incomplete and circular. Clauses 4 through 8 incrementally complete these definitions by starting at a very high level, progressively detailing a set of conceptual models, and describing how they collectively interact to control batch production.

Clause 4, *Batch processes and equipment,* is normative. The intent of this clause is to provide models and terminology that describe batch processes and the equipment used to perform them.

Clause 5, *Structure for batch control,* is normative. The intent is to describe three types of control used in batch processing and their relationships to the previously defined process and equipment models.

Clause 6, *Recipes and procedural elements,* is normative. The intent is to describe the roles and contents of four types of recipes used in batch manufacturing, their use of the previously defined process and procedural control models, and their connection to equipment control.

Clause 7, *Batch control considerations,* is normative. The intent is to describe additional considerations related to iterative design, exception handling, modes and states, production plans and schedules, and production information.

Clause 8, *Activities and functions in batch control,* is normative. The intent is to describe the control activities that are needed to address the diverse control requirements of batch manufacturing.

Clause 9, *Completeness, compliance, and conformance,,* is normative. The intent is to define compliance and conformance relative to the normative models and terminology in this standard.

Annex A is informative. It provides guidance towards understanding the model types used in this standard.

Annex B is informative. It provides a quick summary of the changes made in this update as compared with the original 1995 standard.

Annex C is informative. It provides answers to typical questions that may arise in applying this standard.

Annex D is informative. It provides a more expansive procedural state reference model. The model found in Clause 7 may be considered a collapsed version of this more general model.

Annex E is informative, giving references to further investigation concerning safety.

This standard (Part 1, Models and Terminology) is intended for those who are:

- involved in designing and/or operating batch manufacturing plants;

- responsible for specifying controls and the associated application programs for batch manufacturing plants; or
- involved in the design and marketing of products in the area of batch control.

This standard provides standard models and terminology for defining the control requirements for batch manufacturing plants. The models and terminology defined in this standard:

- emphasize good practices for the design and operation of batch manufacturing plants;
- can be used to improve control of batch manufacturing plants; and
- can be applied regardless of the degree of automation.

This standard provides standard terminology and a consistent set of concepts and models for batch manufacturing plants and batch control that are intended to improve communications between all parties involved, and to:

- reduce the user's time to reach full production levels for new products;
- enable vendors to supply appropriate tools for implementing batch control;
- enable users to better identify their needs;
- make recipe development straightforward enough to be accomplished without the services of a control systems engineer;
- reduce the cost of automating batch processes; and
- reduce life-cycle engineering efforts.

It is important to note that although Clause 3 of this part of the standard provides definitions, the entire document constitutes the models and terminology of batch control. The user should consider Clause 3 as a short glossary of terms with brief descriptions and not rely on Clause 3 for a full understanding of the concepts. The full context of the terms will be found in the body of this standard.

Get more FREE standards from Standard Sharing Group and our chats

It is not the intent of this standard to

- suggest that there is only one way to implement or apply batch control;
- force users to abandon their current way of dealing with their batch processes; or
- restrict development in the area of batch control.

The key concepts defined in this standard are:

- identification of structure and format for recipes and procedures;
- definition of levels of recipes and procedures;
- recognition of product specific recipes and procedures that are separate from process oriented equipment and its direct control;
- identification of a hierarchy of manufacturing equipment and its dedicated control;
- recognition of equipment capabilities that are utilized during recipe and procedure driven production; and
- recognition of the need for modular and re-usable control functionality.

The models presented in this standard are presumed to be complete as indicated. However, they may be collapsed and expanded as described in the explanation of each model.

The series of batch control standards has several parts, as shown in Figure 1. This Part 1 standard focuses on the definitions of process cells and units, master and control recipes, recipe coordination control, and recipe procedural control. Other parts of the series have different focus areas which cover other aspects of batch manufacturing, from the product definitions at enterprises and sites to equipment control within units, equipment modules, and control modules.

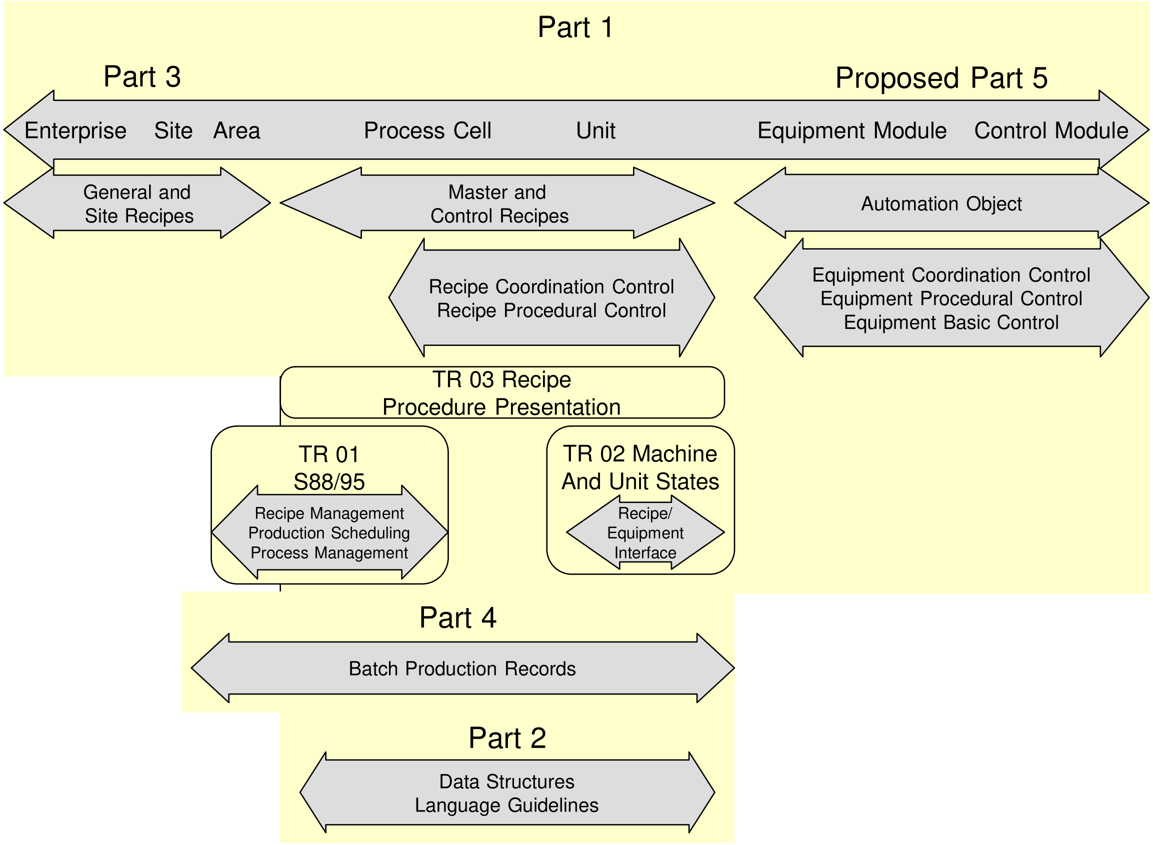


Figure 1 — Standards in the batch control series

This page intentionally left blank.

Get more FREE standards from Standard Sharing Group and our chats

Batch Control Part 1: Models and terminology

1 Scope

This Part 1 standard on Batch Control defines reference models for batch and related procedure-oriented manufacturing as used in the process industries, and terminology that helps explain the relationships between these models and terms. Conformance criteria to this standard are defined in Clause 9.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All normative documents are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. ISA, ANSI, IEC and ISO maintain registers of currently valid normative documents.

- IEC 60848: 2002, GRAFCET specification language for sequential function charts
- IEC 60050-351: 2006, International Electrotechnical Vocabulary – Part 351: Control technology.
- ANSI/ISA-95.00.01-2010 (IEC 62264-1 Mod), Enterprise-Control System Integration – Part 1: Models and Terminology
- ANSI/ISA-95.00.02-2010 (IEC 62264-2 Mod), Enterprise-Control System Integration – Part 2: Object Model Attributes
- IEC/ISO 62264-1, Enterprise-Control System Integration - Part 1: Models and Terminology
- IEC/ISO 62264-2, Enterprise-Control System Integration - Part 2: Object Model Attributes
- ANSI/ISA-18.2-2009, Management of Alarm Systems for the Process Industries

3 Terms, definitions and abbreviations

3.1 Terms and definitions

3.1.1 Introduction

For the purposes of this document, the following terms and definitions apply. The definitions supplied in this clause should be considered a summary statement for the associated term. Since this part of the standard defines models and terminology as a whole, the reader should consider all clauses for the full concept intended for any terms or definition called out in this clause.

3.1.2
allocation
a form of coordination control that assigns a resource or part of a resource to a batch, a procedural element, or an equipment entity as needed

3.1.3
arbitration
a form of coordination control that determines how a resource should be allocated when there are more requests for the resource than can be accommodated at one time

3.1.4
area
a component of a manufacturing site that is identified by physical, geographical, operational, or logical segmentation within the site

NOTE An area may contain process cells, units, equipment modules, and control modules.

Get more FREE standards from Standard Sharing Group and our chats

3.1.5
basic control
the control dedicated to establishing and maintaining a specific state or behavior of equipment and process

NOTE Basic control may include regulatory control, interlocking, monitoring, exception handling, and discrete or sequential control necessary for establishing or maintaining a specific state or behavior.

3.1.6
batch
1) the material that is being produced or that has been produced by a single execution of a batch process;
2) an entity that represents the production of a material at any point in the process;
3) an entity that represents the execution of a control recipe.

NOTE Batch means both the material made by and during the process and also an entity that represents the production of that material. Batch is used as an abstract contraction of the words "the production of a batch."

3.1.7
batch control
control activities and functions that direct batch processes

3.1.8
batch process
a process that leads to the production of finite quantities of material by subjecting quantities of input materials to an ordered set of processing activities over a finite period of time using one or more pieces of equipment

3.1.9
batch schedule
a list of batches to be produced in a specific process cell

NOTE The batch schedule typically contains such information as what is to be produced, how much is to be produced, when or in what order the batches are to be produced, and what equipment is to be used.

3.1.10
campaign
a collection of related batches for scheduling purposes

3.1.11
common resource
a resource that can provide services to one or more requesters

NOTE Common resources are identified as either exclusive-use resources or shared-use resources.

3.1.12
control activity model
a conceptual model identifying seven interdependent activities (several of which are subdivided into functions) that manage control definition, operation, and information for batch processes

3.1.13
control module
the lowest level grouping of equipment in the physical model that can carry out basic control

NOTE 1 This term applies to both the physical equipment and the equipment entity.

NOTE 2 The control module level may not be omitted from the physical model.

NOTE 3 The control module level contains the interfaces to the physical equipment.

3.1.14
control recipe
a type of recipe which, through its execution, defines the manufacture of a single batch of a specific product

NOTE The control recipe may not be omitted from the recipe types model.

3.1.15
coordination control
a type of control that directs, initiates, and/or modifies the execution of procedural control and the utilization of equipment entities

3.1.16
enterprise
an organization that coordinates the operation of one or more sites

3.1.17
equipment control
the equipment-specific functionality that provides the actual control capability for an equipment entity, including procedural, basic, and coordination control

3.1.18
equipment entity
a collection of physical processing and control equipment and equipment control grouped together to perform a certain control function or set of control functions

3.1.19
equipment entity model
a hierarchical model to logically organize the physical assets used in a batch process in combination with the control present at each level

3.1.20
equipment module
a functional group of equipment that can carry out a finite number of specific minor processing activities

NOTE 1 An equipment module is typically centered around a piece of process equipment (a weigh tank, a process heater, a scrubber, etc.). This term applies to both the physical equipment and the equipment entity.

NOTE 2 Examples of minor process activities are dosing and weighing.

NOTE 3 Two types of equipment modules are identified: recipe-aware equipment modules and generic equipment modules.

NOTE 4 Any reference to an equipment module in this standard is understood to be a general reference to both the recipe-aware equipment module and the generic equipment module unless otherwise specified.

Get more FREE standards from Standard Sharing Group and our chats

3.1.21
equipment operation
an operation that is part of equipment control

3.1.22
equipment phase
a phase that is part of equipment control

3.1.23
equipment procedure
the top level procedural element in the procedural control model that is part of equipment control

3.1.24
equipment unit procedure
a unit procedure that is part of equipment control

3.1.25
exception handling
those functions that deal with plant or process contingencies and other events which occur outside the normal or desired behavior of batch control

3.1.26
exclusive-use resource
a common resource that only one user can use at any given time

3.1.27
formula
a category of recipe information that includes process inputs, process parameters, and process outputs

3.1.28
general recipe
a type of recipe that expresses equipment and site independent processing requirements

3.1.29
general recipe procedure model
a conceptual model for general and site recipe procedures that structurally conforms to the process model

3.1.30
generic equipment module
an equipment module which may be initiated through execution of equipment control but is not capable of being directly initiated through the execution of a recipe

3.1.31
header
information about the purpose, source and version of the recipe such as recipe and product identification, creator, and issue date

3.1.32
lot
a unique amount of material having a set of common traits

NOTE Some examples of common traits are material source, the master recipe used to produce the material, and distinct physical properties.

NOTE 2 As defined in ANSI/ISA-95 and IEC 62264-1 as *material lot*: uniquely identifiable amount of a material.

3.1.33
master recipe
a type of recipe that accounts for equipment capabilities and may include process cell-specific information

NOTE 1 The master recipe may not be omitted from the recipe types model.

3.1.34
master recipe procedure model
a conceptual model for master and control recipe procedures that structurally conforms to the procedural control model

3.1.35
mode
the manner in which the transition of sequential functions are carried out within a procedural element or the accessibility for manipulating the states of equipment entities manually or by other types of control

3.1.36
operating schemes
mutually exclusive process actions executed by the same equipment or control module

NOTE 1 Examples for operating schemes are Inner-Temperature / Jacket-Temperature / Delta Temperature Control for a heating/cooling equipment module or Venting / Nitrogen Purge / Evacuation / High Pressure Control for a pressure system equipment module.

NOTE 2 Operating schemes may be implemented in an equipment module entity by alternative branches within one single equipment phase or by multiple mutually exclusive equipment phases of which only one can be active at any given time.

3.1.37
operation
a procedural element defining an independent processing task to accomplish all or part of a process operation, typically specifying the initiation, organization, and control of phases

3.1.38
path
the order of equipment within a process cell that is used in the production of a specific batch

3.1.39
personnel and environmental protection
the control activity that

1) prevents events from occurring that would cause the process to react in a manner that would jeopardize personnel safety and/or harm the environment; and/or
2) takes additional measures, such as starting standby equipment, to prevent an abnormal condition from proceeding to a more undesirable state that would jeopardize personnel safety and/or harm the environment; and/or
3) issues notification of an abnormal condition.

NOTE Personnel and environmental protection are outside of the scope of this part of the standard and is included for completeness in understanding and not to define the required content.

Get more FREE standards from Standard Sharing Group and our chats

3.1.40
phase
the lowest level procedural element in the procedural control model that is intended to accomplish all or part of a process action

3.1.41
physical model
a hierarchical model to logically organize the physical assets of a manufacturing enterprise

3.1.42
procedural control
the type of control that executes a procedure

3.1.43
procedural control model
a hierarchical model which depicts the orchestration of procedural elements to carry out process-oriented tasks

3.1.44
procedural element
a building block for procedural control that is defined by the procedural control model

NOTE This general definition is applied to all levels of the procedural hierarchy in the definitions for procedure, unit procedure, operation, and phase.

3.1.45
procedure

1) a specification of a sequence of steps, actions or activities with a defined beginning and end that is intended to accomplish a specific objective or task
2) the highest level procedural element within the procedural control model, which defines the required set of processing activities for a single batch, typically through the initiation, organization, and control of unit procedures

NOTE Such a procedure may define a process that does not result in the production of product, such as a clean-in-place procedure.

3.1.46
process

1) a sequence of chemical, physical, or biological activities for the conversion, transport, or storage of material or energy
2) a sequence of chemical, physical, or biological activities that change the condition of equipment. An example would be a clean-in-place process

3.1.47
process action
a minor processing task that may be combined with other minor processing activities to make up a process operation

NOTE Process actions are the lowest level of processing task within the process model.

3.1.48
process cell
a logical grouping of equipment that includes the equipment required for production of one or more batches.

NOTE This term applies to both the physical equipment and the equipment entity.

3.1.49
process cell management
the control activity that includes the functions needed to manage batch production within a process cell

3.1.50
process control
the control activity that includes the functions needed to provide sequential, regulatory, and discrete control and to gather and display data

3.1.51
process input
an identification and quantity of materials, energy, or other resources required for a recipe

3.1.52
process model
a hierarchical model which illustrates the subdivision of a batch process

3.1.53
process operation
a major processing task that usually results in a chemical or physical change in the material being processed and that is defined without consideration of the actual target equipment configuration

3.1.54
process output
an identification and quantity of materials, energy, or other resources resulting or expected to result from one execution of a control recipe

3.1.55
process parameter
information that is needed to manufacture a material but does not fall into the classification of process input or process output

NOTE Examples of process parameter information are temperature, pressure, and time.

3.1.56
process stage
a part of a process that usually operates independently from other process stages and that usually results in a planned sequence of chemical or physical changes in the material being processed

3.1.57
recipe
the necessary set of information that uniquely defines the production requirements for a specific product or operational task

NOTE There are four types of recipes defined in this standard: general, site, master, and control.

3.1.58
recipe-aware equipment module
an equipment module containing one or more equipment phases that is capable of being initiated directly through the execution of a recipe

Get more FREE standards from Standard Sharing Group and our chats

3.1.59
recipe management
the control activity that includes the functions needed to create, store, and maintain general, site, and master recipes

3.1.60
recipe operation
the part of a recipe which either defines the sequencing and ordering of recipe phases, or references an equipment operation

3.1.61
recipe phase
the part of a recipe which defines the references to an equipment phase

3.1.62
recipe procedure

1) a general reference to any one of the four levels of the Procedural Element Model applied in the recipe
2) the part of a recipe which either defines the sequencing and ordering of recipe unit procedures, or references an equipment procedure

NOTE A recipe procedure is intended to provide product specific flexibility beyond parameter and formula changes, without making equipment modifications.

3.1.63
recipe types model
a conceptual model identifying the relationships between the types of recipes that are defined in an enterprise

3.1.64
recipe unit procedure
the part of a recipe which either defines the sequencing and ordering of recipe operations, or references an equipment unit procedure

3.1.65
resource
See ANSI/ISA-95 and IEC 62264-1.

NOTE This standard focuses on equipment resources. Other resource types defined in ANSI/ISA-95 and IEC 62264-1 are for the most part not considered

3.1.66
shared-use resource
a common resource that can be used by more than one user at a time

3.1.67
site
a component of a manufacturing enterprise that is identified by physical, geographical, operational, or logical segmentation within the enterprise

NOTE A site may contain areas, process cells, units, equipment modules, and control modules.

3.1.68
site recipe
a type of recipe that is site specific

NOTE Site recipes may be derived from general recipes recognizing local constraints, such as language and available raw materials.

NOTE Site recipes may include routing information necessary for coordinating manufacturing across process cells and production scheduling activities.

3.1.69
state
the condition of an equipment entity or of a procedural element at a given time

NOTE The number of possible states and their names vary for equipment and for procedural elements.

3.1.70
train
a collection of one or more units and associated lower level equipment groupings that has the ability to be used to make a batch of material

3.1.71
unit
a collection of associated equipment modules and/or control modules that can carry out one or more major processing activities

NOTE Examples of major processing activities are react, crystallize, and make a solution.

3.1.72
unit procedure
a strategy for carrying out a major processing task within a unit to accomplish all or part of a process stage typically through the initiation, organization, and control of operations.

3.1.73
unit recipe
the part of a control recipe that uniquely defines the contiguous production requirements for a unit

NOTE The unit recipe contains the unit procedural element and its related formula, header, equipment requirements, and other information.

3.1.74
unit supervision
the control activity that includes functions needed to supervise the unit and the unit's resources

3.2 Abbreviations

For the purposes of this standard, the following abbreviations apply.

ID Identification (A unique identifier for batches, lots, operators, technicians, and raw materials.)

Get more FREE standards from Standard Sharing Group and our chats

4 Batch processes and equipment

4.1 Introduction

The models and terminology defined in this clause provide a foundation for understanding the application of batch control based on this standard. Specifically, this clause provides

- an overview of continuous, discrete, and batch manufacturing
- hierarchical models of equipment groupings (physical model) and functional equipment entities (equipment entity model) for use in batch manufacturing
- criteria for classifying batch manufacturing facilities

4.2 Types of manufacturing

4.2.1 Introduction

A process is a sequence of chemical, physical or biological activities for the preparation, conversion, transport or storage of material or energy. Industrial manufacturing processes can generally be classified as either process manufacturing or discrete parts manufacturing. Process manufacturing can generally be further classified as continuous, batch, or some combination of the two. How a process is classified depends on whether the output from the process appears in a continuous flow (continuous), in finite quantities of parts (discrete parts manufacturing), or in finite quantities of material (batches).

4.2.2 Continuous process manufacturing

In continuous process manufacturing, materials are passed in a continuous flow through processing equipment. Once established in a steady operating state, the nature of the process is not dependent on the length of time of operation. Start-ups, transitions, and shutdowns are usually treated as separate activities and do not necessarily contribute to achieving the desired processing.

The concepts of batch processing in this standard may be applied to continuous processing for such control as start-ups, grade changes, and shutdowns. Minor modifications to the models in this standard could be applied to address these unique processing needs.

EXAMPLE When a grade change occurs, a new product or recipe may need to be started without stopping material flows through the equipment.

4.2.3 Discrete parts manufacturing

In discrete parts manufacturing, a specified quantity of parts moves as a unit (group of parts) between workstations and each part maintains its unique identity. Products are classified into production lots that are based on common raw materials, production requirements, and production histories.

The concepts of batch processing in this standard may be applied to discrete parts manufacturing for such control as start-ups, change-overs, and shutdowns. Minor modifications to the models in this standard could be applied to address these unique processing needs.

4.2.4 Batch process manufacturing

The batch process manufacturing addressed in this standard leads to the production of finite quantities of material (batches) by subjecting quantities of input materials to a defined order of processing actions using one or more pieces of equipment. Both a single execution of a batch process and the product or intermediate material it produces are referred to as a batch. Batch processes are discontinuous processes, which are neither discrete nor continuous but have characteristics of both.

The models in this standard facilitate batch process manufacturing by

- providing a consistent structure for design and operation of batch processing facilities
- separating recipes that define product-specific processing sequences from equipment control that implements fixed equipment-specific processing capabilities
- segmenting both recipes and equipment control based on modular hierarchies of relatively simple and reusable entity structures
- describing a set of required activities and functions (activity model) for
  - initiating recipes based on a batch schedule
  - interpreting the active recipes' defined processing tasks and executing them through equipment control
  - managing recipe and production data

4.3 Process model

4.3.1 Introduction

The process model is a multi-level hierarchical model for subdivision of a batch process and is the basis for defining equipment independent recipe procedures. Each individual element in the model is a procedure, whose directly subordinate elements (if any) comprise its steps. The ultimate goal of the hierarchy is to efficiently organize the processing activities that are specified (either directly or by referencing an external requirement) at its lowest level.

Get more FREE standards from Standard Sharing Group and our chats

The process model levels are illustrated in Figure 2 and described below based on use of the full model. Clause 4.3.6 describes how the process model may be collapsed or expanded.

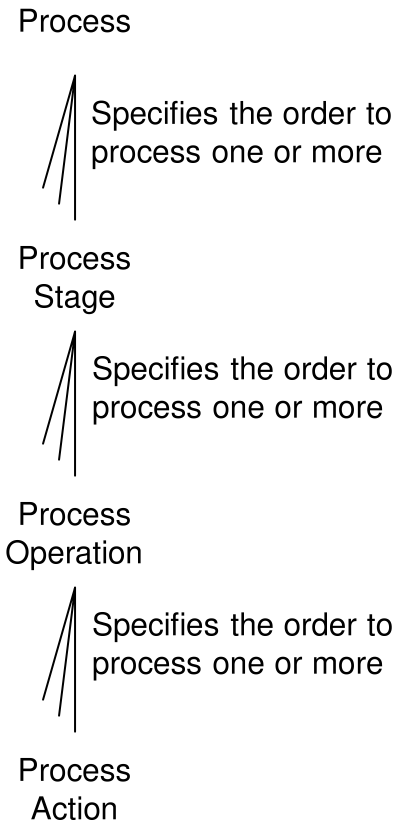

Process
Specifies the order to process one or more
Process Stage
Specifies the order to process one or more
Process Operation
Specifies the order to process one or more
Process Action

Figure 2 — Process model (instance diagram) when not collapsed or expanded

4.3.2 Process

The order of processing for an entire batch is called the process. When the full model is used, the process shall specify the order to process one or more process stages, which may be in series, in parallel, or a combination of both.

EXAMPLE The production of polyvinyl chloride by polymerization of vinyl chloride monomer is used in this clause as an example batch process.

4.3.3 Process stage

A process stage is a major part of a process that usually operates independently from other process stages. It usually results in a planned sequence of chemical or physical changes in the material being processed. When the full model is used, each process stage shall specify the order to process one or more process operations, which may be in series, in parallel, or a combination of both.

EXAMPLE Typical process stages in the polyvinyl chloride process might be the following:

- Polymerize: Polymerize vinyl chloride monomer into polyvinyl chloride.
- Recover: Recover residual vinyl chloride monomer.
- Dry: Dry polyvinyl chloride.

4.3.4 Process operation

Process operations represent major processing activities. A process operation usually results in a chemical, biological or physical change in the material being processed. The processing associated with the term “unit operation” used in chemical engineering often relates to a process operation in the process model. When the full model is used, each process operation shall specify the order to process one or more process actions, which may be in series, in parallel, or a combination of both.

EXAMPLE Typical process operations for the polymerization of vinyl chloride monomer into polyvinyl chloride process stage might be the following:

- Prepare reactor: Evacuate the reactor to remove oxygen.
- Charge: Add demineralized water and surfactants.
- React: Add vinyl chloride monomer and catalyst, heat to 55 - 60ºC, and hold at this temperature until the reactor pressure decreases.

4.3.5 Process action

Process actions represent minor processing activities. The scope of a process action is usually small enough that the specification of its activities (with appropriate variants, such as target values) is highly reusable from one process to another. When the full model is used, the process actions shall specify the required processing activities, either directly or by referencing an external requirement.

EXAMPLE Typical process actions for the react process operation might be the following:

- Add: Add the required amount of catalyst to the reactor.

Get more FREE standards from Standard Sharing Group and our chats

- Add: Add the required amount of vinyl chloride monomer to the reactor.
- Heat: Heat the reactor contents to 55 - 60ºC.
- Hold: Hold the reactor contents at 55 - 60ºC until the reactor pressure decreases.

4.3.6 Collapsing and expanding the process model

The process model presented is complete as indicated and the levels defined shall not be rearranged, but may be collapsed or expanded to suit different processing requirements. Collapsing in this context means omitting one or more defined levels of the process model. Expanding in this context means inserting an additional level into the model.

It shall be permissible to collapse the process model in an implementation if the following requirements are satisfied

- The process level shall not be omitted.
- If a level is not used and one or more lower levels still remain
  - — the next higher implemented level shall take over its procedural functions, including specification of ordering logic for the next lower level.
  - — the procedural functions of the lower level(s) shall not be impacted.
- The lowest implemented level of the process model should specify the required process functionality, either directly or by referencing an external requirement.

Expanding the model shall be restricted to allowing one or more additional levels in the model

- Between process and process stage
- Between process stage and process operation
- Between process operation and process action

Any additional levels must be structurally consistent with the process model and shall not use names defined in this standard.

4.4 Equipment and equipment control

4.4.1 Introduction

The concept of equipment capabilities and usage of these capabilities to accomplish desired processing tasks is a major point of this standard. Equipment control is the equipment-specific control functionality that supports such capabilities and that may be utilized by recipes to carry out their specified processing tasks and by operators to perform direct manipulation of control settings.

An equipment entity comprises a grouping of physical equipment and associated equipment control that has been engineered to perform a certain process or control function or set of such functions. Physical equipment in this context includes processing equipment, instrumentation, and control equipment.

The notion of equipment control being part of an equipment entity is to be understood as an abstract concept. The association may be physical or logical. It is not a statement of the physical implementation of equipment control. However, it should be possible to associate equipment control with a particular equipment entity. The control part of any equipment entity can be satisfied by human actions, automation or a combination of both.

This interaction of equipment control and physical equipment is purposely described without any reference to language or implementation. The intent is to describe a framework within which equipment control and physical equipment may be defined and discussed.

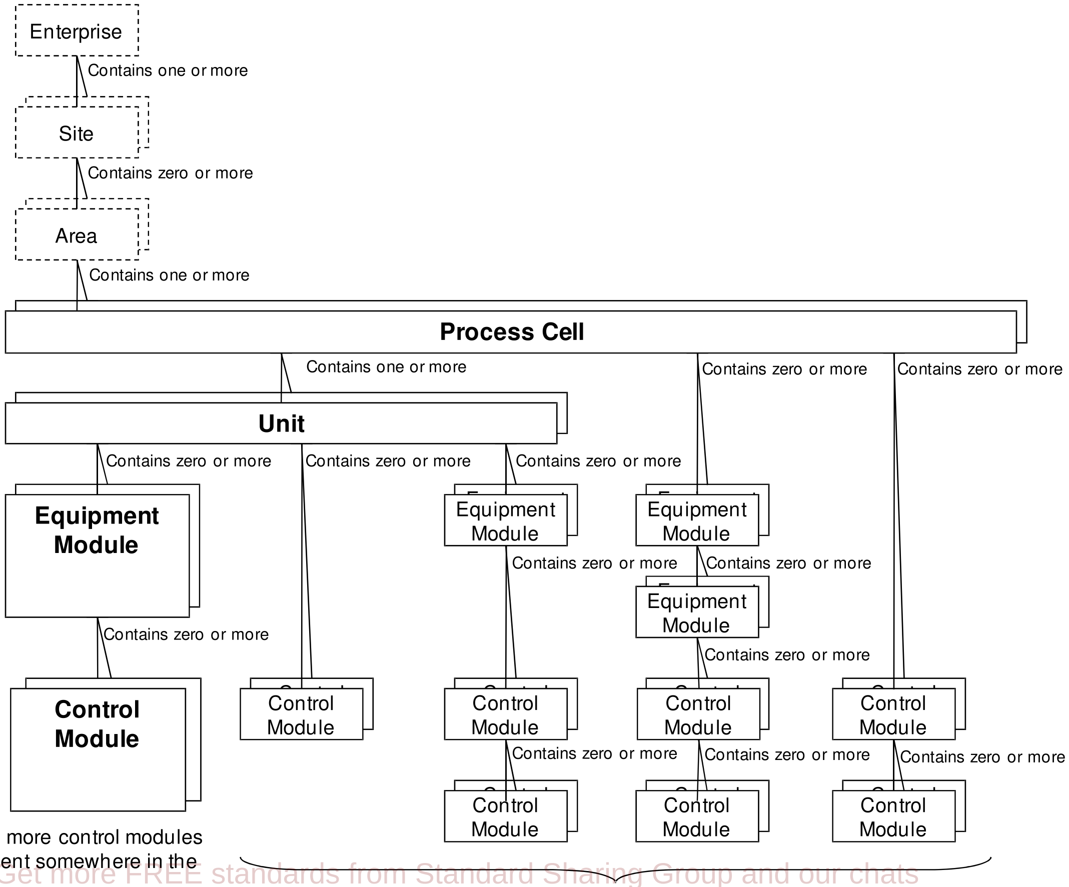
4.4.2 Physical model

4.4.2.1 Introduction

The physical equipment groupings defined for equipment entities, as well as their relationship to a manufacturing enterprise, may be described in terms of a physical model that hierarchically organizes the enterprise into sites, areas, process cells, units, equipment modules, and control modules. Figure 3 illustrates how lower level groupings are combined to form higher levels in this hierarchy in batch process manufacturing.

The enterprise, site, and area levels are more precisely defined in the IEC/ISO 62264 and ANSI/ISA-95 standards. These levels in Figure 3 are shown with dashed lines to indicate that they are often beyond the scope of batch control and criteria for configuring their boundaries are not defined in this standard.

Each equipment grouping comprising a process cell, unit, equipment module, or control module corresponds to a single equipment entity and its boundaries are engineered to satisfy the functional requirements for that entity. The defining characteristics for these levels in the physical model, therefore, correspond exactly to the levels of the equipment entity model and are described more fully in the following discussion of equipment entities in Clause 4.4.3.



Note: One or more control modules must be present somewhere in the process cell.

Some of the other possible relationships

Get more FREE standards from Standard Sharing Group and our chats

Figure 3 — Physical model

4.4.2.2 Enterprise

An enterprise is a collection of one or more sites and their subordinate areas, process cells, units, equipment modules, and control modules.

Personnel responsible for this level generally determine what products will be manufactured, at which sites they will be manufactured, and in general how they will be manufactured.

There are many factors other than batch control that affect the boundaries of an enterprise. Therefore, the criteria for configuring the boundaries of an enterprise are not covered in this standard.

4.4.2.3 Site

A site is a physical, geographical, operational, or logical subdivision of the enterprise. It may contain areas and their subordinate process cells, units, equipment modules, and control modules.

The boundaries of a site are usually based on organizational or business criteria as opposed to technical criteria. There are many factors other than batch control that affect these boundaries. Therefore, the criteria for configuring the boundaries of a site are not covered in this standard.

4.4.2.4 Area

An area is a physical, geographical, operational or logical subdivision of a site. In batch manufacturing, it shall contain one or more process cells and their subordinate units, equipment modules, and control modules.

The boundaries of an area are usually based on organizational or business criteria as opposed to technical criteria. There are many factors other than batch control that affect these boundaries. Therefore, the criteria for configuring the boundaries of an area are not covered in this standard.

4.4.2.5 Collapsing and expanding the physical model

The physical model presented is complete as indicated and the levels defined shall not be rearranged, but may be collapsed or expanded. Collapsing in this context means omitting one or more defined levels of the physical model. Expanding in this context means inserting an additional level into the model.

When using a collapsed or expanded physical model, the levels defined in this standard shall not be rearranged, the site level shall not be omitted. Below the process cell level, the rules described in Clause 4.4.3.7 for collapsing and expanding the equipment entity model also apply to the physical model.

4.4.3 Equipment entity model

4.4.3.1 Introduction

This clause discusses equipment entities that are formed from the combination of equipment control and physical equipment. The equipment entity model (see Figure 4) hierarchically organizes equipment entities based on the lower four equipment groupings of the physical model shown in Figure 3: the process cell and its included units, equipment modules, and control modules.

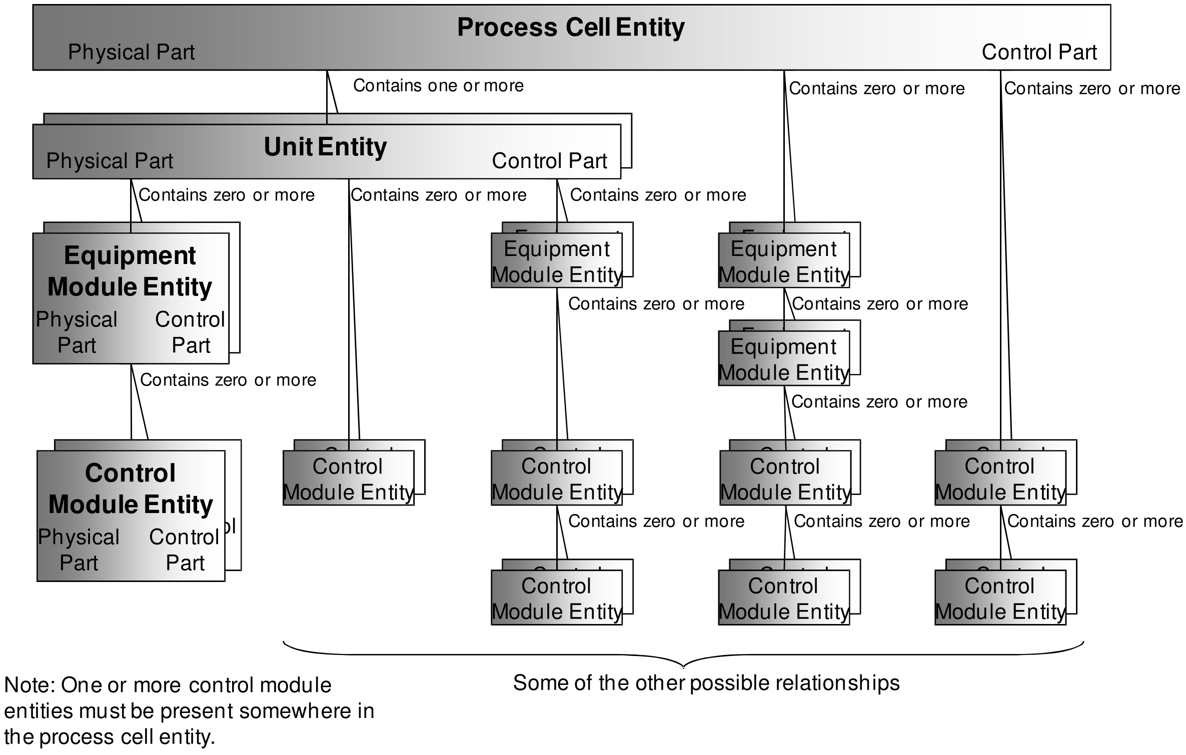


Figure 4 — Equipment entity model

Get more FREE standards from Standard Sharing Group and our chats

These groupings are created to simplify equipment operation by clearly defining the overall functions of each entity and treating it as a single larger piece of controlled equipment. Once created, the equipment groupings in the equipment entity model cannot be split up except by re-engineering the equipment entities.

The general functionality associated with each level of the equipment entity model is described below. The types of control needed for batch manufacturing and their usage within each level are discussed in Clause 5.

In the equipment module and control module levels, an equipment entity may be directly contained either in a larger grouping at the same level (for which the unrestricted level of nesting is not depicted) or in a higher level grouping. Inclusion of process control equipment (sensors and actuators) shall be exclusively at the control module level.

When the terms process cell, unit, equipment module, and control module are used, they most often refer to the equipment entity, not just the physical equipment. Whether equipment control in an equipment entity is implemented manually or by way of automation, it is only through the exercise of equipment control that the equipment can produce a batch.

All control related clauses of the standard assume that each process cell entity has been subdivided into well defined units, equipment modules, and control modules. Effective subdivision of the process cell into well defined equipment entities is a complex task, highly dependent on the individual requirements of the specific environment in which the batch process exists. Inconsistent or inappropriate equipment subdivisions can compromise the effectiveness of the modular approach to recipes suggested by this standard.

Subdivision of the process cell requires a clear understanding of the purpose of the process cell's equipment. Such understanding allows the identification of equipment entities that should work together to serve an identifiable processing purpose.

The control capability possible in the different equipment entities are important characteristics and a main basis for classification of equipment entities. In the following paragraphs equipment control for the individual equipment entities is discussed.

This clause discusses some general principles involved in segmenting a process cell into equipment entities that can carry out specified processing tasks or equipment-specific actions. A more exhaustive explanation of process segmentation principles is beyond the scope of this standard.

It is important to note that the physical process cell design can greatly influence the implementation of batch control. Minor differences in the physical system can dramatically affect the organization of equipment entities and procedural elements.

The subdivision of a process cell should follow the principles listed below:

- The function any equipment entity serves in product processing is clear and unambiguous.
- The function performed by the equipment entity is consistent in terms of processing task, and should be usable for that task no matter what product is being manufactured at a given time.
- Equipment entities within a process cell are clearly defined, including processing capabilities, relationships with other equipment entities and allowable equipment interconnections.
- Subordinate equipment entities are able to execute their task(s) independently and asynchronously, allowing a higher level equipment entity to orchestrate the activities of its subordinates.
- Interactions between equipment entities are minimized. While planned interaction is periodically necessary, each equipment entity should perform its functions while influencing the functioning of other equipment entities as little as possible.
- Equipment entities have clear boundaries.
- A consistent basis for the definition of equipment entities. An operator subsequently interacting with similar equipment entities should be able to do so naturally and without confusion.
- Necessary interaction between equipment entities is, insofar as possible, coordinated by equipment entities at the same level or at the next higher level.

NOTE Portable equipment entities may not have a permanent relationship to one specific process cell, but are allowed to be moved between process cells to improve the flexibility of the facility. In general, moving equipment entities from one process cell to another should only occur when the equipment is not acquired or allocated to a batch.

4.4.3.2 Process cell entity

In the physical model, a process cell is an engineered equipment grouping within an area. In both the physical and equipment entity models, each process cell shall contain one or more units and their subordinate equipment modules and control modules. It may also directly contain physical processing equipment, equipment modules, and control modules that are not part of any unit.

The boundaries of a process cell entity shall be engineered to include the full set of physical processing equipment, units, equipment modules, control modules, and additional equipment control (as described in Clause 5) that will be used to produce batches of one or more products.

The process cell as described in this standard corresponds to a type of Work Center defined in IEC/ISO 62264 and ANSI/ISA-95. A Work Center may be specialized for different manufacturing types and functions. More than one type of Work Center, such as a Storage Zone or a Production Line is often required in an area or site. Unless specifically identified otherwise, the Work Center referred to in this standard as a “Process Cell” is a batch process cell and does not refer to the other types of work centers identified in IEC/ISO 62264 and ANSI/ISA-95.

The existence of the process cell allows for production scheduling on a process cell basis. It also allows for process cell-wide control strategies to be designed, such as for emergency response to process conditions or to comply with administrative requirements such as documentation for good manufacturing practices.

All batch processing tasks are orchestrated at the process cell level based on recipes containing procedure, parameter, and other information and a batch schedule containing operational requirements for each batch. The process cell management activity described in Clause 8.6 performs this orchestration by interpreting the schedule and top level procedural elements of the recipes it specifies.

A frequently recognized subdivision of a process cell is the train. A train is composed of all units and other equipment that (a) make up a recognizable subset of a process cell and (b) is sufficient to produce all batches for one or more of the process cell’s recipes. Any given batch may use only part of the equipment in a train or process cell. Furthermore, more than one batch (including batches for more than one product) may be active in a train or process cell simultaneously. Although a process cell may contain multiple (possibly overlapping) trains, no train may contain equipment outside the boundaries of the process cell.

Get more FREE standards from Standard Sharing Group and our chats

NOTE A train is not considered an additional level in the physical hierarchy.

The process cell will also need to prepare and monitor equipment or resources not currently involved in batch processing, such as which units are available, what units and piping are going through a clean-in-place (CIP) routine, and what the current inventories of raw materials are. This could be initiated by internal logic in the process cell itself or through the batch schedule and recipes.

The complexity of control within the process cell will depend on the equipment available within the process cell, the interconnectivity among this equipment, the degree of freedom of movements of batches through this equipment, and the arbitration of the use of this equipment so that the equipment can be used most effectively.

Equipment modules and control modules may exist as separate entities under direct control of the process cell.

4.4.3.3 Unit entity

A unit entity is an engineered subdivision of a process cell. It may contain physical processing equipment, equipment modules, and control modules.

Each unit shall be engineered to combine all necessary physical processing equipment, equipment modules, control modules, and additional equipment control (as described in Clause 5) required to perform one or more major processing tasks, such as react, crystallize, and make a solution. It is usually centered on a relatively independent major piece of processing equipment, such as a mixing tank or reactor. The complete set of required equipment may be entirely contained within the unit boundaries or may include common use equipment modules or control modules within the same process cell that can be acquired temporarily by the unit to carry out specific tasks.

The unit as described in this standard corresponds to a type of Work Unit defined in IEC/ISO 62264 and ANSI/ISA-95. A Work Unit may be specialized for different manufacturing types and functions. More than one type of Work Unit, such as a Storage Unit or a (Continuous Process) Unit may even be required in a batch oriented Process Cell. Unless specifically identified otherwise, the Work Unit referred to in this standard as a “unit” is a batch processing unit and does not refer to the other types of work units identified in IEC/ISO 62264 and ANSI/ISA-95.

In a batch manufacturing process, a unit may contain or operate on up to one complete batch of material at some point in the processing sequence of that batch. This generally results in a physical separation between batches that is in many cases also a business requirement. However, in other types of work units the separation is not as clear and a logical boundary between batches is used. This is often found in hybrid batch/continuous or batch/discrete (e.g., packaging) processes, in which both types of process equipment are used within a process cell. In such processes, it is not unusual for two or more batches to be within a unit at the same time.

NOTE In continuous and discrete processes, that apply this model, a unit may operate on different products entering and exiting at the same time. This requires appropriate control in the unit to monitor, track, and control the logical separation between the different products.

Units coordinate the functions of the lower level entities, such as equipment modules and control modules. The primary purpose of equipment control in a unit is to control the processing of the batch that is currently associated with the unit.

The definition of a unit’s boundaries and control functions requires knowledge of the major processing tasks it will perform, as well as the innate processing capabilities of the equipment itself. The following guidelines apply:

- One or more major processing tasks, such as reaction or crystallization, may take place in a unit.
- Units should be defined such that they operate independently of each other.
- Units operate on only one batch at a time.

4.4.3.4 Equipment module entity

An equipment module entity is an engineered subdivision of a process cell, a unit, or another equipment module. When contained directly in a process cell, it is usually a common use resource (see Clause 4.4.3.6) that may be acquired temporarily to carry out a specific task. It may contain physical processing equipment, other equipment modules, and control modules.

NOTE When an equipment module contains another equipment module, it may be referred to as a compound equipment module. A compound equipment module is not considered an additional level in the physical hierarchy.

Each equipment module shall be engineered to combine the necessary physical processing equipment, other equipment modules, control modules, and additional equipment control (as described in Clause 5) required to perform one or multiple minor processing tasks or process actions. Equipment modules may be engineered around processing equipment such as a heating/cooling system, an agitator, a pressure and venting system of a unit or auxiliary equipment such as a dosing system. Multiple mutually exclusive process actions which can be performed on the same equipment module are called operating schemes.

NOTE Examples for operating schemes of a heating/cooling system may be Inner-Temperature Control, Jacket-Temperature Control and Delta Temperature Control. Operating schemes for a venting system may be Evacuation, Nitrogen-gassing, and High Pressure Control.

The complete set of required equipment may be entirely contained within the equipment module boundaries or may include common use equipment modules or control modules within the same process cell that can be acquired temporarily by the equipment module to carry out specific tasks.

4.4.3.5 Control module entity

A control module entity is an engineered subdivision of a process cell, a unit, an equipment module, or another control module. When contained directly in a process cell, it is usually a common use resource (see Clause 4.4.3.6) that may be acquired temporarily to carry out a specific task. It may contain control equipment (sensors and actuators) and other control modules.

NOTE When a control module contains another control module, it may be referred to as a compound control module. A compound control module is not considered an additional level in the physical hierarchy.

Each control module shall be engineered to combine the necessary sensors, actuators, other control modules, and additional equipment control (as described in Clause 5) that are required to perform some basic control function. The complete set of required equipment may be entirely contained within the control module boundaries or may include common use control modules within the same process cell that can be acquired temporarily by the control module to execute specific functions.

EXAMPLE Some examples of control modules are

- an individual sensor or actuator

Get more FREE standards from Standard Sharing Group and our chats

- a regulating loop consisting of a transmitter, a controller, and a control valve that is operated via a set point signal
- state-oriented equipment that consists of an on/off automatic block valve with position feedback switches, that is operated via the set point of the equipment
- a header that contains several on/off automatic block valves that coordinates the valves to direct flow to one or several destinations based upon the set point directed to the header control module

A control module can direct commands to actuators if they have been configured as part of the control module. A control module can also direct commands to other control modules if they are contained or in some way referenced by that control module. Control of the process is affected through the equipment specific manipulation of control modules and actuators.

EXAMPLE Some examples of equipment control in control modules include

- opening or closing a valve, with confirmation failure alarms;
- regulating the position of a control valve based on a sensor reading and PID control algorithm;
- setting and maintaining the state of several valves in a material header. This could be a single control module configured to contain all of the valves directly or a compound control module configured to contain several control modules, each of which is configured to directly contain a single valve.

4.4.3.6 Common resources

If more than one requestor (which may be an equipment entity, a batch, or an operator) can acquire or request the services of a single equipment entity, then that equipment entity is designated as a common resource. Common resources are often present within complex batch processes. Common resources are often implemented as either equipment modules or control modules. A common resource may be either exclusive-use or shared-use.

NOTE This standard focuses on equipment entity resources. Other resource types defined in ANSI/ISA-95 and IEC 62264-1 are for the most part not considered.

If the resource is designated as exclusive-use, only one requestor may use the resource at a time.

EXAMPLE A shared weigh tank in a batch plant might be an example of an exclusive-use resource. It can be used by only one reactor at a time. The schedule or some other basis for assignment of its services must take this exclusive-use resource into consideration. If a reactor is waiting for the use of the weigh tank while another is using it, the waiting reactor is idle and is not making product, which has a negative effect on equipment utilization.

If the common resource is designated as shared-use, several requestors may use the resource at the same time.

EXAMPLE Some shared-use resources in a batch plant might be a process heater serving multiple units at the same time or a raw material distribution system which is capable of delivering material to more than one unit at a time.

If the capabilities of a shared-use resource are limited, then it is possible that the requests for service might exceed the capacity of the resource. In that case some of the same concerns about allocation which apply to exclusive-use resources also apply to shared-use resources. Care must also be taken so that one requestor does not improperly shut off or deactivate a resource while other requestors are using it.

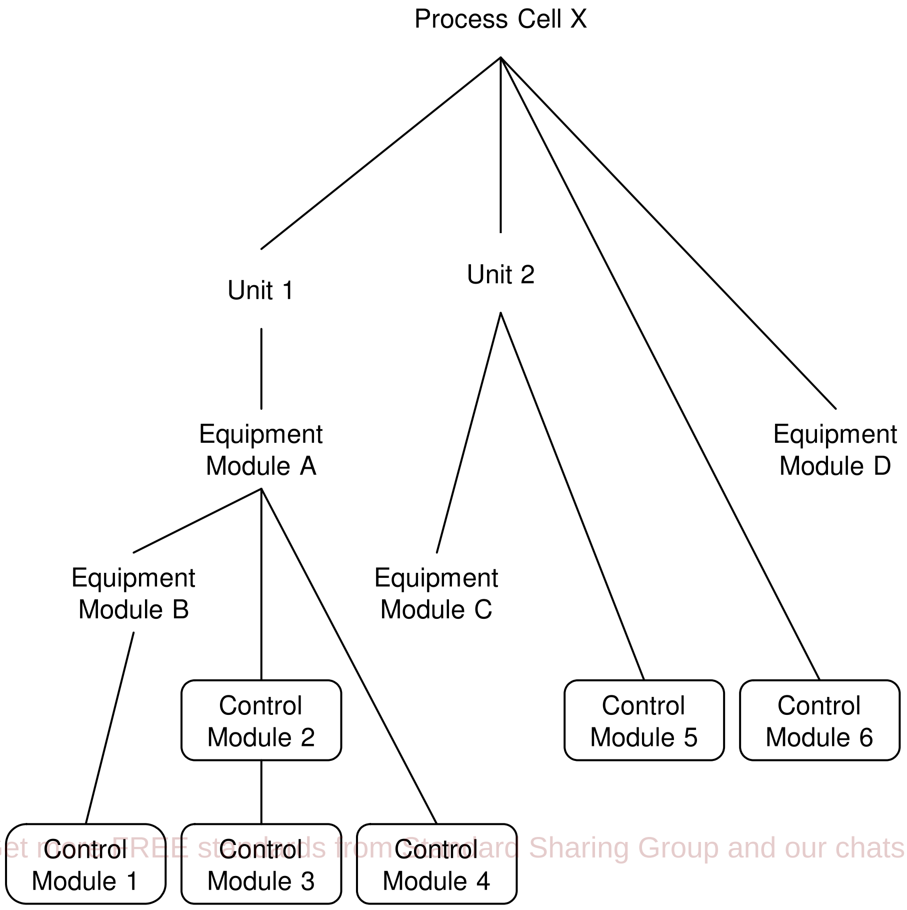
4.4.3.7 Collapsing and expanding the equipment entity model

The equipment entity model presented is complete as indicated and the levels defined shall not be rearranged, but may be collapsed or expanded to suit different processing requirements. Collapsing in this context means omitting one or more defined levels of the equipment entity model. Expanding in this context means inserting an additional level into the model.

In an implementation, subdivisions of each process cell or unit may directly contain any non-overlapping lower level equipment groupings, however, process cells may not be omitted and each must contain at least one unit entity and in some way a control module entity (either directly or indirectly). Since the physical assets of batch processing facilities vary greatly, the application of the equipment entity model shall be flexible and allow for any combination of lower level subdivisions of each process cell, unit, and equipment module.

Expanding the model below the process cell level shall be restricted to allowing one or more additional levels in the model between the process cell and unit, however, they shall be structurally consistent with the physical model and shall not use names defined in this standard.

EXAMPLE Figure 5 illustrates some expected equipment entity architectures that could be found in typical batch environments. The example shows that in some cases the equipment module level may be collapsed (i.e. Unit 2 contains Control Module 5 directly).



Get more FREE standards from Standard Sharing Group and our chats

Figure 5 — Equipment entity model examples

4.5 Process cell classification

4.5.1 Introduction

This clause discusses the classification of process cells by the number of different products manufactured in the process cell and by the physical structure of the equipment used in the manufacturing.

4.5.2 Classification by number of products

A process cell is classified as single-product or multi-product based on the number of products planned for production in that process cell.

A single product process cell produces the same product in each batch. Variations in procedures and parameters are possible. For example, variations may occur in order to compensate for differences in equipment, to compensate for substitute raw materials, to compensate for changes in environmental conditions, or to optimize the process.

A multi-product process cell produces different products utilizing different methods of production or control. There are three possibilities:

- All products are produced with the same procedure using different formula values (varying materials and/or process parameters). Sometimes products produced in this way are referred to as grades of a product.
- The products are produced using different procedures.
- The products produced are grouped into families, each sharing a unique procedure among its products, but providing different formula values for each product.

4.5.3 Classification by physical structure

The order of equipment actually used or expected to be used by a specific batch is called the path. A process cell is classified as single path, multiple path, or network based on its physical structure. Regardless of which structure is used, several batches may be in progress at the same time (in different units), multiple input materials may be used, multiple finished materials may be generated, and units may share input material sources and product storage.

A single-path structure is a group of units through which a batch passes sequentially (see Figure 6). A single-path structure could be a single unit, such as a reactor, or several units in sequence.

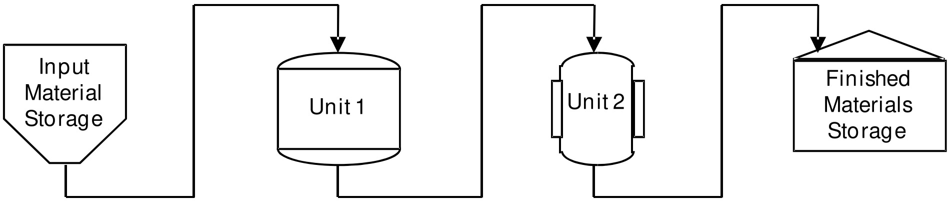


Figure 6 — Single-path structure

A multiple-path structure is shown in Figure 7. It consists of multiple single-path structures in parallel with no product transfer between them. Although units within a multi-path structure may be physically similar, it is possible to have paths and units that are of radically different physical design.

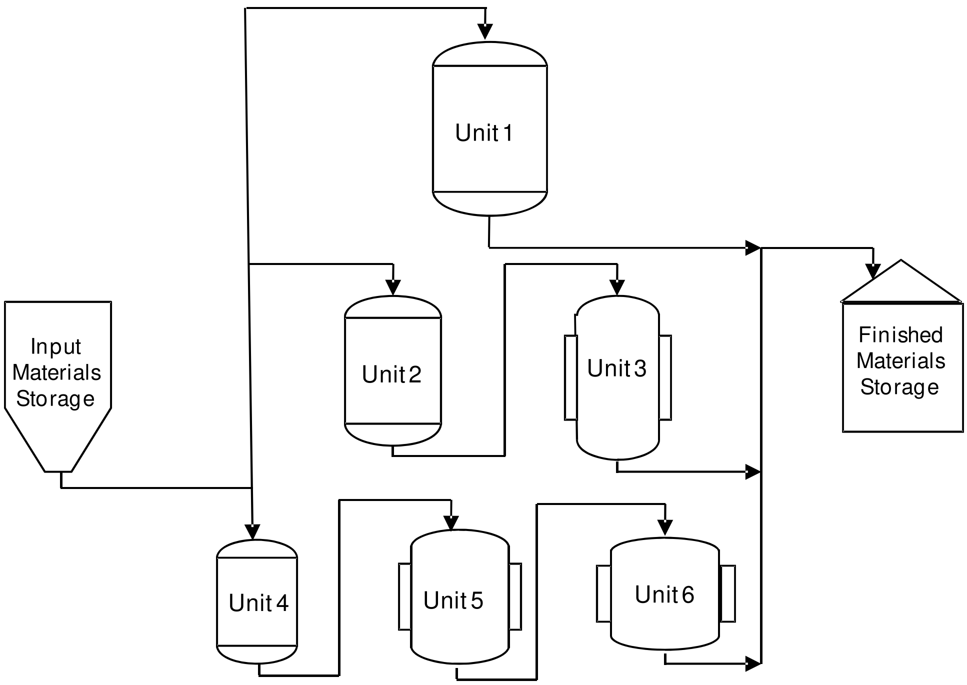


Figure 7 — Multiple-path structure

A network structure is shown in Figure 8. The paths may be either fixed or variable. When the paths are fixed, the same units are used in the same sequence. When the path is variable, the sequence may be determined at the beginning of the batch or it may be determined as the batch is being produced. The path could also be totally flexible. For example, a batch would not have to start at either Unit 1 or Unit 3; it could start with any unit and take multiple paths through the process cell. The units themselves may be portable within the process cell. In this case, verification of the process connections may be an important part of the procedures.

Get more FREE standards from Standard Sharing Group and our chats

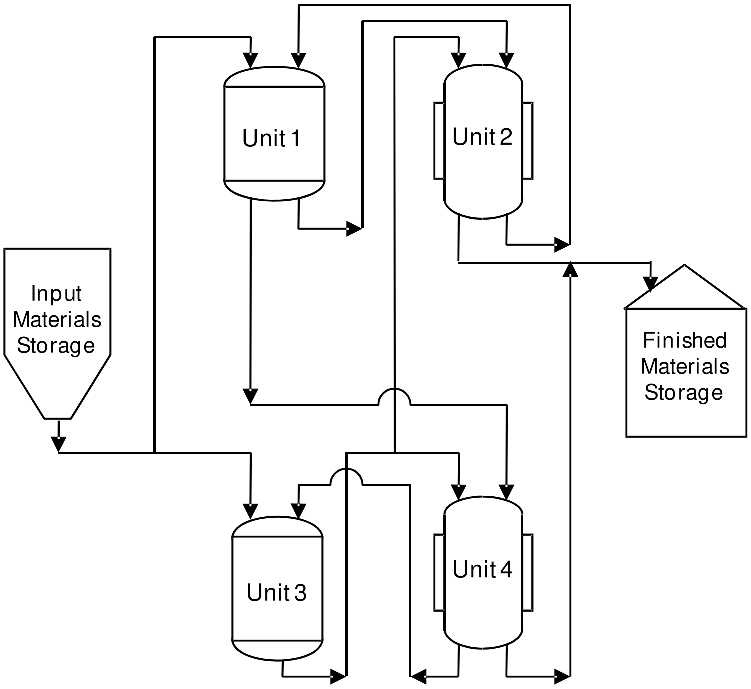


Figure 8 — Network structure

These are three possible structures for a process cell, actual implementations can vary widely. In addition, the structure of the process cell may include or exclude material storage.

5 Structure for batch control

5.1 Introduction

This clause describes

- three types of control (basic control, procedural control, and coordination control) typically needed to provide process functionality and control capabilities for batch manufacturing
- the relationship of equipment entities to each type of control
- the forms of coordination control referred to as allocation and arbitration
- the relationship of both the process model and the equipment entity model to the procedural control model.

The controls described may be implemented entirely through automation, through human intervention, or through a combination of both.

5.2 Basic control

5.2.1 Introduction

Basic control comprises a subset of equipment control that is dedicated to establishing and maintaining a specific state or behavior of equipment and process. Basic control

- includes regulatory control, interlocking, monitoring, exception handling, and discrete or sequential control necessary for establishing or maintaining a specific state or behavior;
- may respond to process conditions that in turn influences the control outputs or triggers corrective actions;
- may be activated, deactivated, or modified by operator commands or by procedural or coordination control;
- exposes equipment and process condition information.

Get more FREE standards from Standard Sharing Group and our chats

NOTE 1 Regulatory control is dedicated to maintaining a process variable or variables at or near some desired value. Complex control strategies such as multivariable control, model-based control, and artificial intelligence techniques also fit into this category of regulatory control.

NOTE 2 State-oriented control refers to setting the state of a piece of equipment as opposed to the state of a process variable or variables. State-oriented equipment has a finite number of states. It defines a product independent processing sequence.

NOTE 3 Basic control strategies may include exception handling logic that is activated when an equipment malfunction or abnormal process deviation occurs.

The actions of basic control in a batch environment are in principle no different from the control of continuous processes. However, in the batch environment, there may be higher requirements on the ability for basic control to receive commands and to modify its behavior based on these commands.

5.2.2 Basic control in equipment entities

5.2.2.1 Introduction

This clause describes the role of basic control in each level of the equipment entity model.

5.2.2.2 Basic control in process cells

Basic control in a process cell is generally performed by equipment modules and control modules that are either directly within the process cell or within units the process cell contains. Equipment modules and control modules directly within the process cell may support basic control that spans or is referenced by several units.

EXAMPLE 1 A main supply valve on a common header line whose position needs to be monitored by multiple units in the process cell.

EXAMPLE 2 A process cell may contain basic control for an interlock that shuts one unit down and propagates the shutdown to upstream units that are feeding this particular unit.

5.2.2.3 Basic control in units

Basic control in a unit is generally performed by equipment modules and control modules that are within the unit or in some way referenced by the unit.

EXAMPLE A rupture disk indicator on a reactor is used to generate an interlock condition on the unit.

5.2.2.4 Basic control in equipment modules

Basic control in an equipment module is generally performed by control modules that it contains or references. Equipment modules may also directly execute basic control. However, direct interaction with physical equipment, such as actuators or sensors, is accomplished through referenced or included control modules.

EXAMPLE 1 Basic control through contained control modules: An equipment module opens a supply valve that is driven by a control module which is subordinate to the equipment module by issuing commands to the valve control module.

EXAMPLE 2 Direct basic control: An equipment module may calculate a temperature compensated set-point for use by other equipment entities through the use of basic control.

EXAMPLE 3 Interaction with equipment through control modules: An equipment module opens a common supply valve that is outside of its defined scope by issuing commands to the referenced control module that operates it. (In this case, the referenced control module is a common resource that must be allocated to the equipment module before issuing such commands.)

5.2.2.5 Basic control in control modules

Control modules may execute basic control.

EXAMPLE The common supply valve of the preceding example is instead automatically opened by the control module if the temperature is within limits and the downstream valve is open. (Allocation is not required in this case because the valve is contained in the control module.)

5.3 Procedural control

5.3.1 Introduction

Procedural control executes procedures, which are characteristic of batch processes. A procedure is a specification of a sequence of steps, actions, or activities with a defined beginning and end that is intended to accomplish a specific process-oriented objective or task.

A procedure consists of any combination of steps in series and/or in parallel. Transition conditions may be inserted between any steps to modify which steps will execute and in what order, usually based on equipment, process, and operator responses. Steps are initiated only after the immediate predecessor(s) in series with them have completed and any intervening transition conditions are true.

There is a difference between the implicit sequences in basic control and the explicit sequences in procedural control. In basic control, the sequence is primarily equipment oriented, is implicit in the design of the basic control element, usually does not vary from one execution of the basic control element to another, and is an inseparable part of setting and maintaining a state or condition in the process.

EXAMPLE A basic control sequence may control the order in which valves open and shut to change a double block and bleed valve assembly from open to closed; another may control the sequence of actions necessary to properly change the speed of a two-speed motor.

Explicit sequences in procedural control are overtly stated. They are not implicit in the design of a control element, may vary from one execution of the procedure to the next, and are intended to achieve a process-oriented result. In procedural control, the sequence causes the execution of a series of planned state or condition changes or a planned and ordered series of process oriented tasks to take place.

EXAMPLE Procedural control includes the sequence of actions necessary to carry out a process oriented task, such as charging a reactor with a specific amount and type of raw material and reporting the result; another higher level procedure may direct the order in which to charge different raw materials and how much of each to charge.

5.3.2 Procedural control model

5.3.2.1 Introduction

The procedural control model is a multi-level hierarchical model to accomplish the task of a complete process (or part of a process) based on the resources of a specific process cell. It is also the basis for defining equipment dependent recipe procedures. Each individual element in the model is a procedure, whose directly subordinate elements (if any) comprise its steps. The ultimate goal of the hierarchy is to efficiently organize the process-oriented tasks that are to be executed (either automatically or manually) at its lowest level.

Get more FREE standards from Standard Sharing Group and our chats

The procedural control model levels are illustrated in Figure 9 and described below based on use of the full model. Clause 5.3.2.6 describes how the procedural control model may be collapsed or expanded.

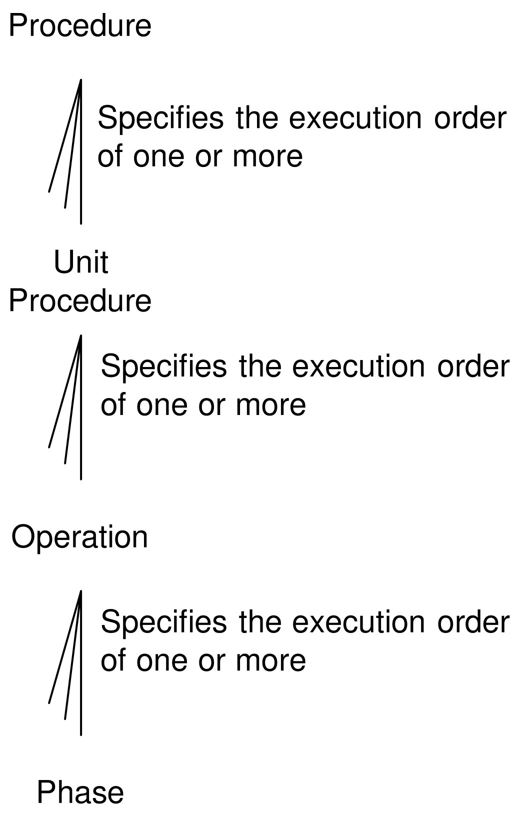

Procedure
Specifies the execution order of one or more
Unit Procedure
Specifies the execution order of one or more
Operation
Specifies the execution order of one or more
Phase

Figure 9 — Procedural control model (instance diagram) when not collapsed or expanded

5.3.2.2 Procedure

The procedure that defines the order of processing for an entire batch represents an alternative, more specific, application of the term procedure. This usage equates to the top level in the procedural control model. When the full model is used, the procedure shall specify the execution order of one or more unit procedures, which may be in series, in parallel, or a combination of both.

NOTE In this context, a procedure is often referred to as a batch procedure.

EXAMPLE An example of a procedure is "Make polyvinyl chloride."

5.3.2.3 Unit procedure

A unit procedure specifies a contiguous production sequence that takes place within a single unit. No more than one unit procedure shall be active in a unit at any time, however, multiple unit procedures specified by one or multiple procedures may run concurrently in different units. When the full model is used, each unit procedure shall specify the execution order of one or more operations, which may be in series, in parallel, or a combination of both.

EXAMPLE Examples of unit procedures include the following:

- Polymerize vinyl chloride monomer.
- Recover residual vinyl chloride monomer.

- Dry polyvinyl chloride.

5.3.2.4 Operation

An operation specifies a major processing sequence, usually to take material being processed from one state to another involving a chemical or physical change. The procedure associated with the term “unit operation” used in chemical engineering often relates to an operation element in the procedural control model. When the full model is used, each operation shall specify the execution order of one or more phases, which may be in series, in parallel, or a combination of both.

An operation is carried to completion in a single unit. Typically only one operation will be active in a unit at given time, but in some cases it may be necessary for multiple operations to be active within the same unit at the same time. Operation boundaries should normally be located at points in the procedure where normal processing can safely be suspended.

EXAMPLE Examples of operations include the following:

- Preparation: Pull a vacuum on the reactor and coat the walls with antifoulant.
- Charge: Add demineralized water and surfactants.
- React: Add vinyl chloride monomer and catalyst, heat, and wait for the reactor pressure to drop.

5.3.2.5 Phase

A phase is the lowest level procedural element in the procedural control model which is either a representation in a recipe or an implementation in equipment control that is intended to accomplish all or part of a process action

Get more FREE standards from Standard Sharing Group and our chats

A phase is the lowest level of procedural control that can accomplish a process-oriented task, however, the set of steps that define its actions are usually equipment specific. The scope of a phase is usually small enough that the specification of its actions (with appropriate variants, such as target values) is highly reusable from one procedure or set of equipment to another. When the full procedural control model is used, phases shall execute the required processing task(s), either automatically or through operator actions. A phase is carried to completion in a single unit or equipment module.

EXAMPLE Examples of phases include the following:

- Add vinyl chloride monomer.
- Add catalyst.
- Ramp and Soak.

The execution of a phase may result in one or more (typically several) of the following actions

- commands to basic control, such as
  - changing controller modes, initializing their outputs, and adjusting their set-points
  - setting, clearing, and changing alarm and other limits
  - modifying algorithm selections and tuning constants
- the collection of
  - process measurement values, such as tank level or liquid density
  - operator entered data

  - values calculated by basic control at the behest of the phase, such as the totalized mass flow into or out of a unit
- locally calculated values, such as the average, minimum, and maximum temperatures during a reaction
- conducting operator interactions
- request action by and exchange data with one or more other procedural elements at the phase level (either in the same or another equipment entity) that are not illustrated as part of the procedural control model

NOTE The ability to request action by and exchange data with other procedural elements is often utilized to

- initiate common-use equipment modules through unit-based phases; or
- enhance the reuse of both types of procedural elements by separating equipment specific execution steps (for example, measuring the amount of a material transfer via loss in weight, incremental level, or totalized flow) from the core process-oriented task definitions and data collection requirements; or
- simplify recipes by executing coordinated actions within the same phase, as may be required for unit initialization or coordination of simultaneous raw material feeds during a reaction sequence

5.3.2.6 Collapsing and expanding the procedural control model

The procedural control model presented is complete as indicated and the levels defined shall not be rearranged, but may be collapsed or expanded to suit different automation requirements. Collapsing in this context means omitting one or more defined levels of the process model. Expanding in this context means inserting an additional level into the model.

It shall be permissible to collapse the process model in an implementation if the following requirements are satisfied

- The procedure level shall not be omitted.
- If a level is not used and one or more lower levels still remain
  - the next higher implemented level shall take over its procedural functions, including specification of ordering logic for the next lower level.
  - the procedural functions of the lower level(s) shall not be impacted.
- The lowest implemented level of the procedural control model should specify the required process functionality, either directly or by referencing an external requirement.

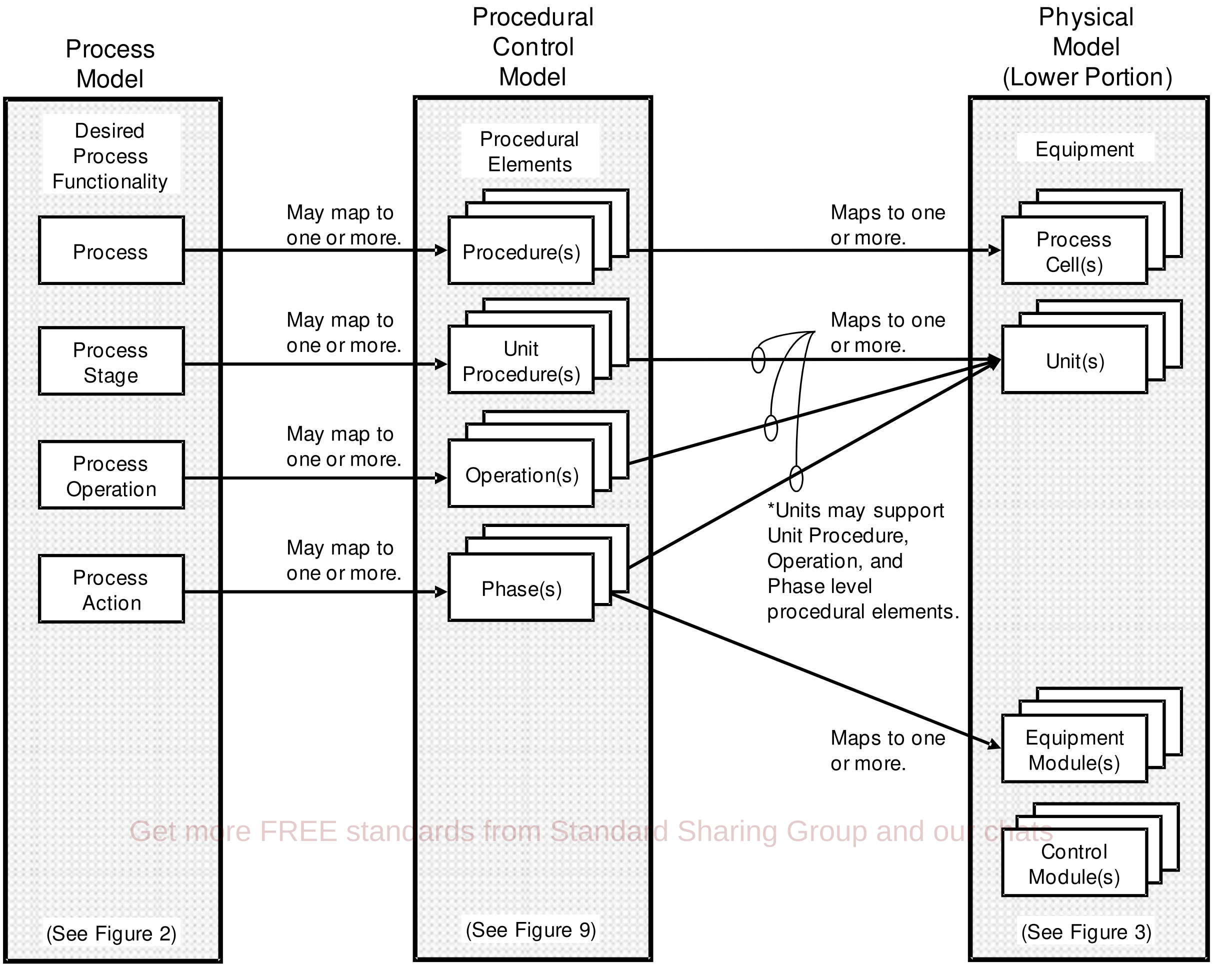
Expanding the model shall be restricted to allowing one or more additional levels in the model

- Between procedure and unit procedure
- Between unit procedure and operation
- Between operation and phase

Any additional levels must be structurally consistent with the process model and shall not use names defined in this standard.

5.3.3 Procedural control model/physical model/process model relationship

The general relationship between the procedural control model, the physical model, and the process model is illustrated in Figure 10. This mapping of procedural control with individual equipment provides processing functionality described in the process model.



Get more FREE standards from Standard Sharing Group and our chats

Figure 10 — Typical process/procedure/equipment mapping to achieve process functionality

The concept of equipment capabilities and usage of these capabilities to accomplish desired processing tasks is a major point of this standard. Through the application of equipment control (composed of procedural control, coordination control, and basic control), the equipment entity may perform the functions defined in procedural elements. Since procedural elements are defined to correspond to elements of the Process Model, the desired processing for a batch is accomplished.

It is important to note that procedural elements may be engineered as part of the equipment entity, or defined as part of the recipe. Equipment entities above the control module level may be engineered to contain elements of defined procedural control, termed *equipment* procedural elements. The combination of equipment control and the defined equipment procedural elements defines an equipment capability which may be referenced by a recipe to accomplish some desired processing task in order to produce a batch.

The equipment procedural elements should be generally designed to be product independent actions which the equipment entity may perform with some pre-engineered adjustments allowing for specific recipe information.

EXAMPLE An equipment procedural element may define how the equipment entity will provide the HEAT capability. The desired temperature which the equipment entity will reach may be a pre-engineered allowance for specific recipe information, e.g. the HEAT temperature set-point in the recipe.

Procedural elements may also be defined within a recipe. These are termed *recipe* procedural elements and may be defined by the recipe author. They are considered to be product dependent. Through the application of equipment control, the equipment entity will perform the functions defined by the recipe procedural element to accomplish a product specific task. This concept allows for a high degree of process flexibility that is essential in many batch processes. It allows for significant changes in process activities to be defined by a recipe author without re-engineering equipment entities.

5.3.4 Procedural control in equipment entities

5.3.4.1 Introduction

Procedural control in equipment entities implements equipment dependent procedures based on the procedural control model and the subordinate phases described in Clause 5.3.2.5. This clause describes the scope of procedural control in each level of the equipment entity model.

5.3.4.2 Procedural control in process cells

The execution of a procedure and initiation of the individual unit procedures is a process cell responsibility. The execution may or may not be integral to the coordination control involved with the movement of batches as described in Clause 5.4.3.2.

5.3.4.3 Procedural control in units

Units may include and execute equipment phases, equipment operations, and equipment unit procedures or they may execute recipe operations and recipe unit procedures passed on to it.

5.3.4.4 Procedural control in equipment modules

Equipment modules may execute equipment procedural control at the phase level, but they do not have the capability of executing higher level procedural elements such as equipment operations. Procedural control in an equipment module may command control modules and other equipment modules within or referenced by that equipment module. It may also issue requests to procedural control in units or other equipment modules. The equipment phase may be capable of initiation automatically or by other means, such as directly by an operator.

Equipment modules may be classified as recipe-aware or as generic, which are distinguishable from one another by whether they can be initiated directly through the execution of a recipe. Both types are optional and may be used together, independently, or not at all. The definitions and expected use of recipe-aware and generic equipment modules are:

1. Each recipe-aware equipment module shall contain one or more equipment phases that are based on the procedural control model and capable of being directly initiated by recipe phases (see Clause 6.6.3). This type of equipment module is utilized in carrying out the intent of recipes when the full procedural control model is used and some or all equipment phases have been encapsulated at this level rather than in units.

2. All other equipment modules are classified as generic equipment modules. They may execute procedural control but this control is not included in the scope of the procedural control model defined in this part of the standard. Equipment modules of this type bear the same relationship to recipes and procedural elements as do control modules. As such, their

composition can be very flexible and they may be commanded at the behest of other equipment modules or of higher level equipment entities.

EXAMPLE An equipment module executing a sequence of actions on the control modules it contains, which is independent of a direct recipe procedure relationship, is an example of a procedural element that may not be associated with a recipe phase.

5.3.4.5 Procedural control in control modules

Control modules do not perform procedural control.

5.4 Coordination control

5.4.1 Introduction

Coordination control directs, initiates, and/or modifies the execution of procedural control and the utilization of resources for batch processing. It is time varying in nature, like procedural control, but it is not structured along specific process-oriented or equipment-oriented tasks.

EXAMPLE Examples of coordination control are algorithms for

- managing resources
- supervising availability and capability (including capacity) of equipment
- allocating equipment to batches
- arbitrating requests for allocation
- coordinating common resource equipment

Get more FREE standards from Standard Sharing Group and our chats

- managing procedural elements
- dynamically binding recipe procedural elements to the proper resources
- managing conditions under which procedural elements may be initiated
- initiating execution of the lowest implemented level procedural elements
- selecting and processing higher level procedural elements
- synchronizing the execution of two or more procedural elements
- managing modes and states, which includes propagating modes and states

The functions that are needed to implement coordination control are discussed in more detail in Clause 8.

5.4.2 Allocation and arbitration

5.4.2.1 Introduction

This clause discusses mechanisms for allocating resources to a batch or unit and for arbitrating the use of common resources when more than one requester needs to use a common resource at the same time.

Resources such as equipment are assigned to a batch or a unit as they are needed to complete or to continue required processing. Allocation is a form of coordination control that makes these assignments. When more than one candidate for allocation exists, a selection algorithm such as “use oldest clean vessel” or "use vessel with lowest duty time" might be used as a basis for choosing the resource. When more than one request for a single resource is made, arbitration is needed to determine which requester will be granted the resource. An algorithm such as "first come/first served" might be used as a basis for arbitration.

In the following clauses, allocation and arbitration are discussed in terms of equipment. The concepts apply equally well to other resources, such as operators.

5.4.2.2 Allocation of equipment entities to batches

The very nature of batch processing requires that many asynchronous activities take place in relative isolation from each other with periodic points of synchronization. Many factors, both expected and unexpected, can affect the time required by one or more of the asynchronous activities from one point of synchronization to the next. For those reasons, and because of the inherent variation in any manufacturing process, the exact equipment which will be available at the time it is needed may be difficult to predict over a significant period of time. Even though a schedule may have been planned to totally optimize the processing sequence from the standpoint of equipment utilization, it often is desirable to allow alternate equipment to be used if the equipment entities planned for a batch are not available when planned. In this case the allocation of equipment entities to the batch -- the routing or path of the batch -- is a decision which should be made every time there is more than one path the batch can take through the available equipment.

5.4.2.3 Arbitration

If there are multiple requesters for a resource, arbitration is required so that proper allocations can be made. Arbitration resolves contention for a resource according to some predetermined algorithm and provides definitive routing or allocation direction. The algorithm may take various forms such as a predetermined schedule with reservations, a batch priority scheme, or it might rely upon operator judgment. Arbitration may bring with it two distinct issues which affect complexity, resource reservation and pre-emption.

Resource reservation allows a claim to be placed on a resource prior to actual allocation. Reservation allows arbitration to be based on future needs rather than allowing the first request for allocation of an idle resource to take precedence regardless of priority. Pre-emption occurs when a higher priority batch or resource is allowed to cancel or interrupt the use of a resource already allocated to a lower priority batch or resource. When allowed, it is most often associated with allocation of exclusive-use common resource but can apply to allocation of any resource.

5.4.3 Coordination control in equipment entities

5.4.3.1 Introduction

This clause describes the role of coordination control in each level of the equipment entity model.

5.4.3.2 Coordination control in process cells

Coordination control in process cells provides the following functions.

- Coordinating the execution of a number of different procedures when the process cell contains multiple units and/or processes multiple batches at the same time.
- Control the movement of batches which may involve a number of choices between alternate paths. Although these choices may be made via links between units, the process cell may also have to determine the routing.
- Provide arbitration to optimize the use of resources, such as shared-use resources and resources that should be reserved well in advance of the time actually needed.

- Coordination control propagates modes to multiple units as needed.

EXAMPLE Examples of coordination control in a process cell include algorithms that

- manage the initialization and movement of the batches being processed within the process cell; and
- initiate and/or associate unit procedures, parameters and other information in individual units in the proper order to cause them to process the product described by the unique combination of schedules and recipes.

5.4.3.3 Coordination control in units

Equipment control in a unit typically includes a substantially higher level of coordination control than any of the lower level equipment entities. This may include algorithms that manage unit and acquired resources; arbitrate requests for services from other units or from the process cell; acquire the services of resources from outside the unit; propagate modes down to equipment modules and up to process cells, and communicate with other equipment entities outside unit boundaries.

5.4.3.4 Coordination control in equipment modules

Coordination control in an equipment module includes coordination of its component parts and may include algorithms for propagating modes and for arbitrating requests.

5.4.3.5 Coordination control in control modules

Coordination control in a control module may include algorithms for propagating modes and for arbitrating requests from units or other modules for usage.

Get more FREE standards from Standard Sharing Group and our chats

6 Recipes and procedural elements

6.1 Introduction

A recipe a collection of information that uniquely defines the manufacturing requirements for a specific product, intermediate or equipment status change such as clean-in-place, sterilize, etc. The concept of recipes allows for a high degree of processing flexibility that is essential in many batch processes and allows for significant changes in process activities to be defined by a recipe author without re-engineering equipment entities.

This clause defines the concept of recipes, four specific types of recipes and how content differs between them. Relationships are established among procedural elements found in recipes and equipment control. The concept of recipe collapsibility is also discussed.

6.2 Recipe types

6.2.1 Introduction

This clause discusses four types of recipes. Recipes provide a way to describe products and how those products are produced. Depending on the specific requirements of an enterprise, other recipe types may exist. However, this standard discusses only the general recipe, site recipe, master recipe, and control recipe shown in Figure 11.

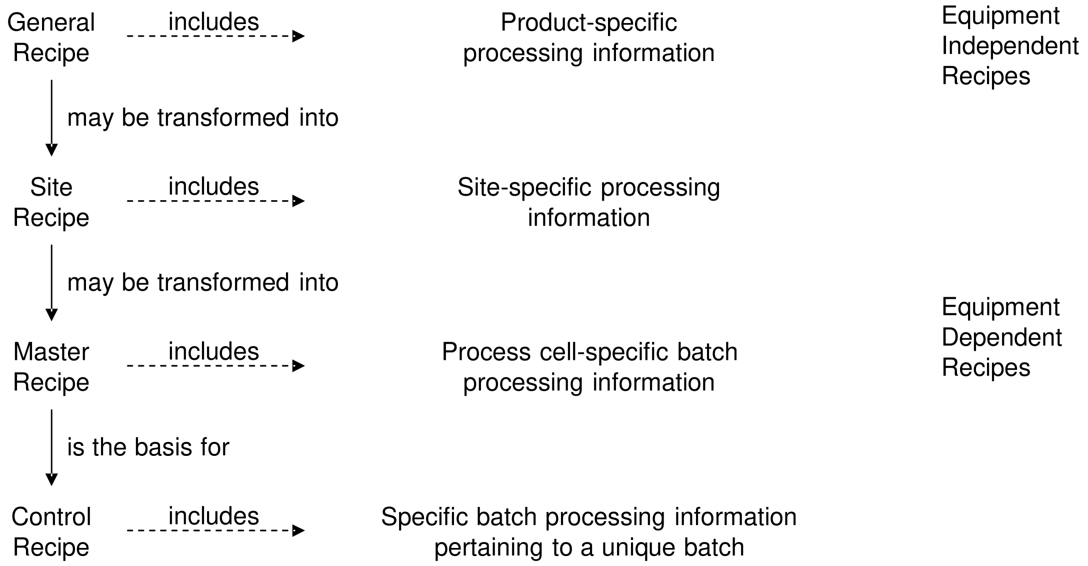

General Recipe
includes
Product-specific processing information
Equipment Independent Recipes
may be transformed into
Site Recipe
includes
Site-specific processing information
may be transformed into
Master Recipe
includes
Process cell-specific batch processing information
Equipment Dependent Recipes
is the basis for
Control Recipe
includes
Specific batch processing information pertaining to a unique batch

Figure 11 — Recipe types model

Fundamental to the practical application of recipes is the concept that different parts of an enterprise may need information about the manufacture of a product in varying degrees of specificity. Because different recipients of the information use it for different purposes, more than one type of a recipe is needed in an enterprise.

The recipe types model presented in this standard is complete as indicated, but may be collapsed or expanded. For example, an enterprise may choose not to implement one or more of the recipe types. The master recipe and the control recipe shall not be omitted from the recipe types model.

A product may be made in many different arrangements of equipment at many different sites. Recipes that are appropriate for one site or set of equipment may not be appropriate for another site or set of equipment. This can result in multiple recipes for a single product. There should be sufficient structure in the definition of recipes to allow tracing of the genealogy of any given recipe.

The recipe does not contain equipment control. The recipe contains process or other procedure-related information for a specific product. This permits batch processing equipment to make many different products without having to redefine equipment control for each product.

There is a substantial difference between general/site recipes and master/control recipes. The general and site recipes are equipment independent and describe the processing technique, that is, how to do it in principle. Master and control recipes are specific to and dependent on the equipment in a defined process cell. They define the procedure that implements the process with actual physical resources.

6.2.2 General recipe

The general recipe is applicable at the enterprise level and serves as the basis for lower-level recipes. The general recipe is created without specific knowledge of or information about the process cell equipment that will be used to manufacture the product. It identifies raw materials, their relative quantities, and required processing, but without specific regard to a particular site or the equipment available at that site. It is created by people with knowledge of both the chemistry and processing requirements peculiar to the product in question and reflects their interests and concerns.

Get more FREE standards from Standard Sharing Group and our chats

While the general recipe is not specific to equipment or to a particular site, the technology for manufacturing a product will usually have evolved sufficiently beyond the laboratory so that equipment requirements can be described in enough detail to allow definition of the type of equipment needed at a particular site or in sites. The general recipe provides a means for communicating processing requirements to multiple manufacturing locations.

Quantities may be expressed as fixed or normalized values. Equipment requirements are expressed in terms of the attributes needed by the equipment, such as pressure requirements and materials of construction.

Although the general recipe contains no scheduling information as such, it may be used as a basis for enterprise-wide planning and investment decisions. It may be part of, or referenced by production specifications and, as such, used for production planning and for information to customers and authorities.

Refer to Part 3 of this standard for further details of general recipes.

6.2.3 Site recipe

The site recipe is specific to a particular site, but is still not specific to a particular set of process cell equipment. It is the combination of site-specific and general recipe information. It may be derived from a general recipe to meet the conditions found at a particular manufacturing location. Alternatively, it may be created directly without the existence of a general recipe. Although it contains no scheduling information as such, it provides much of the processing detail necessary for site-level, long-term production scheduling. A site recipe accommodates such things as the language in which it is written or local raw material or policy differences as site-specific variances. It is still not specific to a particular set of process cell equipment. Typically, the site recipe is the output of a local "site focused" process development function.

There may be multiple site recipes derived from a general recipe, each covering the requirements of different sites or a part of the general recipe in the case of a product that is implemented partly in one site and partly in another.

Refer to Part 3 of this standard for further details of site recipes.

6.2.4 Master recipe

The master recipe is specific to the equipment, raw materials and capabilities of a process cell or a subset of process cell equipment. At the master recipe level, for example, it is essential that there be alignment between the recipe and the capabilities of equipment such as materials of construction, agitation, heating, connections to and from external services or other units, etc. A master recipe may be derived from general or site recipe information. Alternately, it may be created as a stand-alone master recipe if the recipe creator has the necessary process and product knowledge.

Some characteristics of master recipes include the following:

- There may be multiple master recipes derived from a site recipe, either to allow manufacture of the product in more than one slightly different process cell or to allow manufacture of a portion of a specified product in one process cell and a portion in another.
- The sequence of and conditions for initiation of each unit procedure or subordinate equipment procedural entity is either specified or can be uniquely inferred so that permitted paths through the process cell can be determined.
- In a master recipe, the formula data may be specified as normalized values, calculated values or fixed values.
- The master recipe is specific to a product but is not specific to a particular batch so it cannot contain timing and detailed scheduling information as such. However it does contain much of the product-specific information required for detailed scheduling, such as process input information and equipment requirements.
- The master recipe is a required level in the recipe types model. Without it no control recipes can be created and, therefore, no batches can be produced.
- Whether the batch manufacturing equipment is operated manually or fully automatically, the master recipe exists as an identifiable set of written or electronic instructions, or as a combination of multiple forms of information.

6.2.5 Control recipe

The control recipe starts as a copy of a specific version of a master recipe and is then modified as necessary with scheduling and operational information to be specific to a single batch. It contains product-specific process information necessary to manufacture a particular batch of a specific product or portion of a product. It provides the level of detail necessary to initiate and monitor equipment procedural elements in a process cell. It may be modified at any time, for example, to account for actual raw material quantities, material properties, the selection of units, or appropriate sizing.

Since modifications of a control recipe can be made over a period of time based on schedule, equipment, and operator information, a control recipe may go through several modifications during the batch processing.

EXAMPLE Examples of control recipe modifications include:

- defining the equipment that will actually be used for the control recipe at the initiation of the batch or when it becomes known;
- adding or adjusting parameters based on an "as-charged" raw material quality or mid-batch analysis;
- changing the procedure or equipment based on some unexpected event.

6.3 Recipe contents

6.3.1 Introduction

All recipes contain the following categories of information: header, formula, equipment requirements, procedure, and other information. The following subparagraphs provide details regarding these categories. Significant changes from one recipe type to another are noted.

6.3.2 Header

The administrative information in the recipe is referred to as the header. Typical header information may include recipe and product identification, version number, change management information, approval information, recipe status, and other administrative information.

EXAMPLE A site recipe header may contain the name and version of the general recipe from which it was derived.

6.3.3 Formula

Get more FREE standards from Standard Sharing Group and our chats

The formula is a category of recipe information that includes process inputs, process parameters, and process outputs.

A process input is the identification and quantity of a raw material or other resource required to make the product. In addition to raw materials which are consumed in the batch process in the manufacture of a product, process inputs may also include energy and other resources such as manpower. Process inputs consist of both the name of the resource and the amount required to make a specific quantity of finished product. Quantities may be specified as absolute or relative values, as equations based upon other formula parameters or as values related to the batch or equipment size. Process inputs may specify allowable substitutions.

A process parameter details information such as temperature, pressure, or time that is pertinent to the product but does not fall into the classification of input or output. Process parameters may be used as set points, comparison values, or as inputs to conditional logic.

A process output is the identification and quantity of a material and/or energy expected to result from one execution of the recipe. This data may detail environmental impact and may also contain other information such as specification of the intended outputs in terms of quantity, labelling, and yield.

The types of formula data are distinguished to provide information to different parts of an enterprise and need to be available without the clutter of processing details.

6.3.4 Equipment requirements

Equipment requirements constrain the choice of the equipment that may eventually be used to implement a specific part of the procedure.

In general and site recipes, the equipment requirements are typically described in general terms, such as allowable materials of construction and required processing characteristics. It is the guidance from and constraints imposed by equipment requirements that will allow the general or site recipe to eventually be used to create a master recipe which targets appropriate specific equipment.

At the master recipe level, the equipment requirements may be expressed in any manner that specifies allowable equipment in process cells. If trains have been defined, then it is possible for the master recipe (and the resulting control recipe) to be based on the equipment of the train rather than the full range of equipment in the process cell.

At the control recipe level, the equipment requirements are the same as, or a subset of, the allowable equipment in the master recipe. The control recipe may include the identification of specific process cell equipment when these selections are known.

6.3.5 Procedure

The recipe procedure defines the strategy for carrying out a process. The general and site recipe procedures are structured using the levels described in the process model which allows the process to be described in non-equipment specific terms. The master and control recipe procedures are structured using the level described in the procedural control model, the elements of which have a relationship to equipment.

At the lowest level of the procedure, the recipe creator is limited to the use of procedural elements that have been defined, engineered or configured in terms of basic process capabilities or equipment procedural control and have been made available for use in creating a recipe procedure. These sets of procedural elements are often implemented as libraries of procedural elements that serve as building blocks for construction of a recipe procedure. Recipe authors may use any combination of these building blocks in any order to define a procedure. Determination of which of these procedural elements may be part of the procedure is an application specific design decision based on many factors including the capabilities of the process and the degrees of freedom appropriate for the recipe creator in a given application.

EXAMPLE A library of recipe phases defines the phases associated with each unit that may be used in creation of a recipe.

6.3.6 Other information

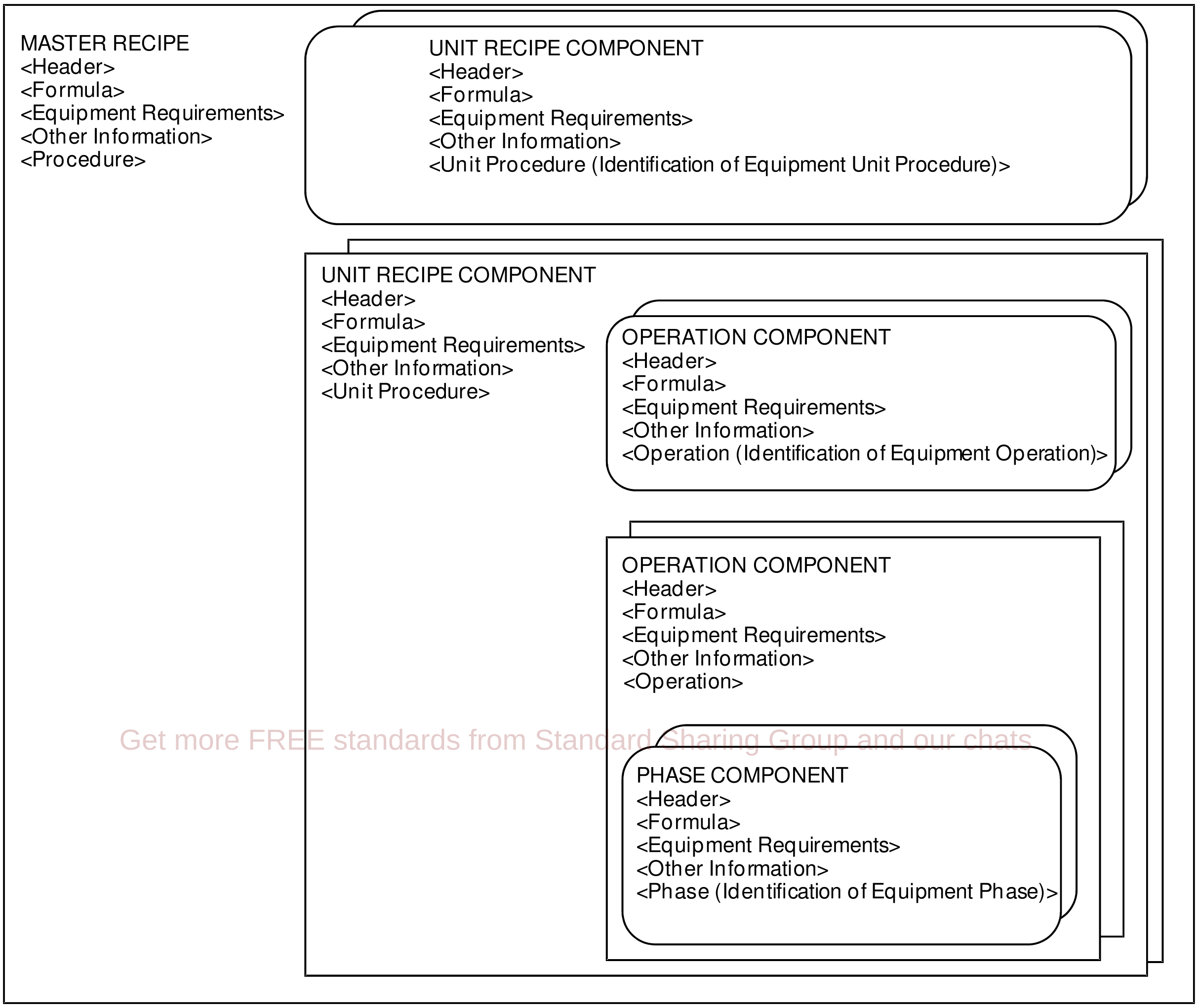
Other information is a category of recipe information that may contain batch processing support information not contained in other parts of the recipe.

EXAMPLE Examples include regulatory compliance information, process safety information, process flow diagrams, and packaging/labelling information.

6.4 Recipe components

Recipes can be viewed as a collection of components. See Part 2 for further information on recipe components.

Figure 12 illustrates how a master recipe could consist of encapsulated recipe components each containing the recipe procedural element appropriate for the level in the hierarchy and the other four defined types of associated recipe information.



Get more FREE standards from Standard Sharing Group and our chats

Figure 12 — Master recipe component encapsulation

The figure illustrates recipe components for two types of procedural relationships:

1) recipe components as recipe procedural elements that directly identify the related equipment procedural elements (rectangle with rounded corners), and

2) recipe components that define the procedure as the ordered set of subordinate recipe procedural elements (rectangle with square corners). Both types of relationships are further discussed in Clause 6.6.

6.5 Recipe procedures by type of recipe

6.5.1 Introduction

This clause describes the procedures used in general, site, master, and control recipes. These recipes and the procedures they contain are of two general categories, equipment independent and equipment dependent.

General and site recipe procedures are equipment independent and are based on the Process Model defined in Clause 4.3. They are process oriented, describing the processing technique to produce a batch – that is, how to do it in principle. An equipment independent procedure contains information that is useful in the design or selection of process cells that will meet its manufacturing requirements, and are useful in the creation of an equipment dependent procedure for carrying out its intent with respect to each process cell.

Master and control recipes are equipment dependent and are based on the Procedural Control Model defined in Clause 5.3.2. Equipment dependent procedures are production oriented, describing the procedure that implements the process with actual resources – that is, how to do it using the equipment in a specific process cell or train. It is the order and content of equipment dependent procedures and their orchestration of basic control that enables equipment to perform batch processing.

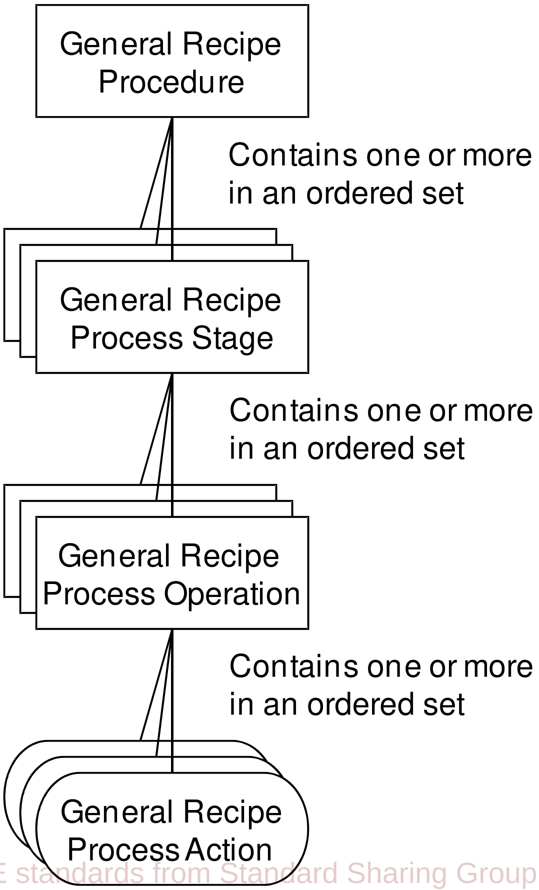
6.5.2 General recipe procedure

The procedure in the general recipe is expressed in three levels of breakdown: Recipe Process Stages, Recipe Process Operations, and Recipe Process Actions (represented by Figure 13). The functionality of these levels corresponds to the functionality of the analogous levels in the Process Model (see Clause 4.3). The general recipe procedure model may be collapsed or expanded in the same manner as described in Clause 4.3.6 for the process model.

The recipe process stage, recipe process operation, and recipe process action are not constrained by unit boundaries in any real plant. They describe processing activities that others may choose to execute in one or in many different units as the general and site recipe is transformed to run in one or more real plants.

A recipe author creates the general recipe procedure by assembling a number of clearly defined and named processing activities in a specific order. The highest three levels in Figure 13 are shown as rectangles to indicate that each level defines a procedure in that they specify the order of subordinate general recipe procedural elements. General Recipe Process Actions are shown as rounded rectangles to indicate that they are not necessarily procedures in and of themselves but are intended to be references to separately defined process actions. The individually defined processing activities should be unambiguous and understood by anyone responsible for creating a recipe at any level.

Creation of a general recipe procedure is a process. If processing activities are defined at the process action level as shown in Figure 13, the proper number of named process actions are assembled in the proper order to create the functionality of one or more general recipe process operations. Likewise, one or more general recipe process stages are created by combining one or more process operations in the proper order and the general recipe procedure is defined in terms of an ordered set of one or more of the previously created process stages. If the processing activities are defined at the process operation or some other level in the general recipe procedure model, the process is similar, but the resulting general recipe procedure is still based on predefined and well understood processing activities.


General Recipe Procedure
Contains one or more in an ordered set
General Recipe Process Stage
Contains one or more in an ordered set
General Recipe Process Operation
Contains one or more in an ordered set
General Recipe Process Action

Get more FREE standards from Standard Sharing Group and our chats

Figure 13 — General recipe procedure model

6.5.3 Site recipe procedure

The procedure information in a site recipe consists of a site recipe procedure that specifies the order of site recipe process stages, site recipe process operations, and site recipe process actions that relate directly to those defined by the general recipe and are, at the lowest level, based on the same processing task definitions. In general, there is a 1:1 correspondence between the recipe process stages in a general recipe and the recipe process stages in a site recipe, between the recipe process operations in a general recipe and the recipe process operations in a site recipe, and between the recipe process actions in a general recipe and the recipe process actions in a site recipe except that, as with other site recipe information, the site recipe process stages, site recipe process operations and site recipe process actions may be modified to make the recipe site-specific.

6.5.4 Master recipe procedure

The recipe procedure portion of the master recipe is made up of ordered sets of recipe unit procedures, recipe operations, and recipe phases (see Figure 14). The master recipe procedure model may be collapsed or expanded in the same manner as described in Clause 6.6.5 for the control recipe procedure model.

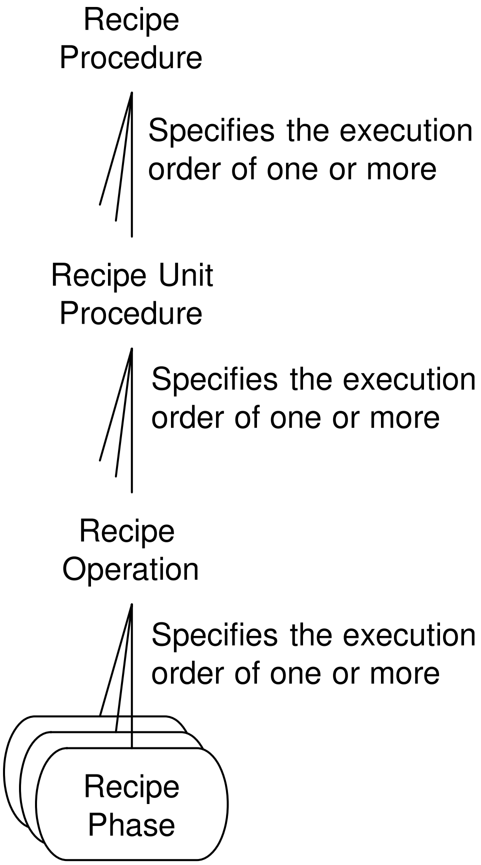

Recipe Procedure
Specifies the execution order of one or more
Recipe Unit Procedure
Specifies the execution order of one or more
Recipe Operation
Specifies the execution order of one or more
Recipe Phase

Figure 14 — Master recipe procedure model

The highest three levels in Figure 14 are shown as rectangles to indicate that each level defines a procedure that specifies the order of subordinate master recipe procedural elements. Master recipe phases are shown as rounded rectangles to indicate that they are not procedures as such but are references to equipment phases that are procedural and are part of the equipment control that is to be used to actually manufacture the defined batch.

If processing tasks are defined at the recipe phase level as shown in Figure 14, the recipe is defined in terms of recipe phases that specifically identify and serve as a reference to an equipment phase that exists in equipment control and has the capability of supporting a linking reference between recipe procedural elements and equipment procedural elements. That capability includes but is not limited to:

- an identification that can be referenced by the recipe procedural element
- the capability of accepting formula and other information associated with the recipe procedural element.

A recipe author or a system that has the capability of transforming a site or general recipe into a master recipe is responsible for creating the master recipe procedure by specifying the identity and the order of recipe unit procedures, recipe operations and recipe phases. Each recipe operation is defined in terms of the predefined recipe phases that have been made available for use in the creation of a master recipe and specify the order in which the corresponding equipment phase should be executed in the equipment. Likewise, each recipe unit procedure is defined as an ordered sequence of one or more recipe operations and the master recipe procedure is defined in terms of an ordered set of one or more master recipe unit procedures. If the references to equipment procedural control are at a level other than the equipment phase, the process is similar, but the resulting master recipe procedure is still based on references to equipment procedural elements.

The creation of a procedure in a master recipe from a procedure in a general or site recipe may be quite complex. It is at this recipe level that the master recipe procedure necessary to carry out the intended process actions, process operations, and process stages must be determined.

Although there is a general similarity between the processing intent of process actions and the processing function defined by recipe phases, there is not necessarily a one-to-one correspondence between the two. One process action may correspond to several recipe phases, and several process actions may correspond to a single recipe phase.

There is a similar relationship between process operations and master recipe operations. There are significant differences also. Master recipe operations target a single unit while process operations are not constrained to units in any specific facility. A single process operation might require one or more master recipe operations to carry out the processing intent described.

There is a similar relationship between process stages and master recipe unit procedures as there is between process operations and master recipe operations. Equipment unit procedures are also carried to completion in a single unit in the target equipment while process stages are not constrained by equipment boundaries in any specific facility. A single process stage might require one or more recipe unit procedures to carry out the processing intent described. Likewise, multiple process stages might be implemented using only one unit procedure. Part 3 of this standard addresses in depth the principles and the process of master recipe procedure creation based on site or master recipes.

Get more FREE standards from Standard Sharing Group and our chats

Although the master recipe procedure does not interact directly with equipment control, the lowest level recipe procedural element (e.g. recipe phase) must refer to an existing and well defined equipment procedural element so that the control recipe that will result from the master recipe has the capability to cause the initiation of the specified equipment procedural element (e.g. equipment phase).

The equipment procedural element referenced by the master recipe procedural element must have:

- an identification that can be referenced by the master recipe procedural element, and
- a means to accept formula and other information associated with the recipe procedural element.

6.5.5 Control recipe procedure

The procedure of a control recipe consists of recipe unit procedures, recipe operations and recipe phases that relate directly to those defined by the master recipe. When the control recipe is created, there is a 1:1 correspondence between all elements of the master recipe procedure and the newly created control recipe procedure. The control recipe procedure may be subsequently modified based on scheduling, process or operational information. Those changes in the control recipe procedure initially or during the execution of a batch may cause it to differ from the master recipe procedure.

In a control recipe, as in a master recipe, the procedure is divided along unit procedure boundaries to provide the process cell with the processing requirements of the recipe on a unit-by-unit basis.

Except at the lowest implemented level of the procedural control model, the procedure must be interpreted at the process cell or unit level. Each step in the defined procedure consists of initiating and awaiting termination of one subordinate procedure. At the lowest implemented level, the recipe procedure directs the execution of an equipment procedural element (see Clause 8.7.1) that may in turn issue directions to operators, subordinate equipment modules, or control modules.

6.6 Control recipe procedure/equipment control relationship

6.6.1 Introduction

This clause discusses the linked information flows. Specifically it addresses:

- the separation between recipe procedural elements in control recipes and equipment entities,
- how recipe procedural elements reference equipment procedural elements,
- how equipment procedural elements command basic control, and
- the rules for collapsing and expanding the procedural control model with regards to connecting recipe procedural elements, equipment procedural elements and equipment entities.

When based on the full master recipe procedure model, any resulting control recipe procedure defines a set of subordinate recipe unit procedures and the procedural order in which they should become active during execution of the batch. Each of those recipe unit procedures defines a set of subordinate recipe operations and their intended procedural order. Each of those subordinate recipe operations defines a subordinate set of recipe phases and their intended procedural order. The recipe phase has no subordinates and is not a procedure itself, but an unambiguous reference to an equipment phase that is part of equipment control in an equipment entity that will carry out the required processing task. It is a link by reference to its corresponding equipment procedural element at the same hierarchical level. A recipe phase shall link by reference only to an equipment procedural element at the phase level.

6.6.2 Linking recipe entities and equipment entities

The control recipe procedure does not directly control equipment. Only equipment control has that ability. The recipe procedure, along with appropriately associated formula, header, equipment requirements and other information, serves as a set of directions that equipment control, primarily coordination control, has the ability to examine or interpret to determine which equipment procedural control activities should be executed and the order in which that should take place.

Figure 15 shows the flow of information from a general recipe to an actual equipment entity that is directed by a control recipe. This figure illustrates the abstract nature of the general, site and master recipes and their recipe procedural elements. The flow of information between recipe types is a transformation. From the control recipe to the equipment entity, and within the equipment entity there is a less abstract and more physical linkage that should be achieved in order for the physical equipment to be operated.

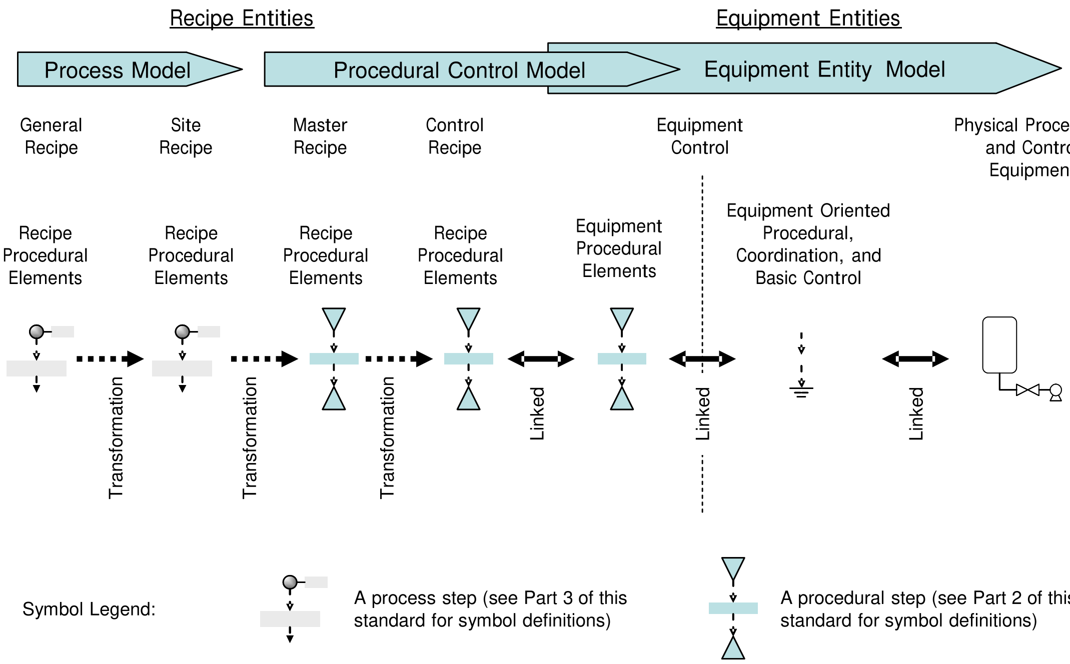

Recipe Entities
Equipment Entities
Process Model
Procedural Control Model
Equipment Entity Model
General Recipe
Site Recipe
Master Recipe
Control Recipe
Equipment Control
Physical Processing and Control Equipment
Recipe Procedural Elements
Recipe Procedural Elements
Recipe Procedural Elements
Recipe Procedural Elements
Equipment Procedural Elements
Equipment Oriented Procedural, Coordination, and Basic Control
Transformation
Transformation
Transformation
Linked
Linked
Linked
Symbol Legend:
A process step (see Part 3 of this standard for symbol definitions)
A procedural step (see Part 2 of this standard for symbol definitions)

Figure 15 — Information flow from general recipe to equipment entity

The linked data flows indicate bi-directional information flow since these reflect operational information flows consisting of commands, parameters, and other information sent from the control recipe toward the physical processing equipment and status and reporting information sent from the physical processing equipment towards the control recipe. The actual information sent in any of the information flows is dependent upon the application.

Get more FREE standards from Standard Sharing Group and our chats

6.6.3 Linking recipe phases and equipment phases

The logical separation of recipes and equipment provides the ability to separate product specific definitions, instructions and information (e.g. recipes) from processing capabilities (e.g. equipment entities). In a given implementation there may or may not be a physical separation that matches this logical one.

EXAMPLE Examples are:

- A control recipe, or parts of it, may be downloaded to an embedded computing system that was supplied by the physical equipment manufacturer and also contains the equipment entity's control equipment.
- A control recipe may run in a dedicated computing system that uses a network connection to communicate with the equipment entity and its equipment control
- Both the control recipe and equipment procedural elements may be implemented in the same computing or control system.
- All equipment interactions are performed manually by a person reading from a written recipe

In all cases the logical separation discussed above exists.

Figure 16 shows the logical separation between control recipe procedural elements and an equipment entity using the complete procedural control model in the control recipe. In Figure 16 recipe phases contain references to equipment phases that are part of a unit or a recipe-aware equipment module.

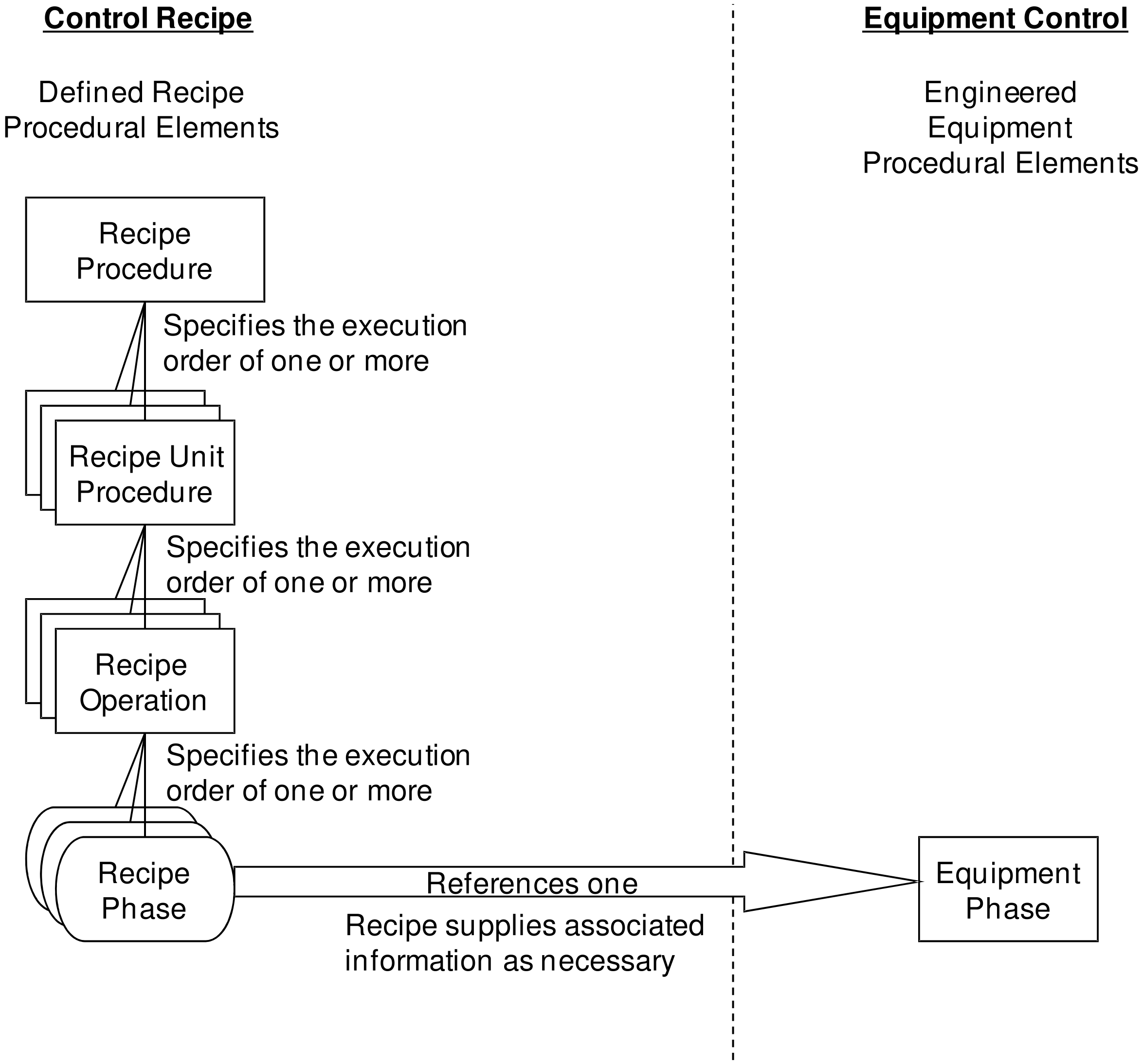

Control Recipe
Defined Recipe Procedural Elements
Recipe Procedure
Specifies the execution order of one or more
Recipe Unit Procedure
Specifies the execution order of one or more
Recipe Operation
Specifies the execution order of one or more
Recipe Phase
References one
Recipe supplies associated information as necessary
Equipment Control
Engineered Equipment Procedural Elements
Equipment Phase

Figure 16 — Control recipe procedure referencing equipment procedural elements at the phase level

Recipe phases, like equipment phases, may have both human and automation system interfaces for receiving inputs such as commands, modes and data as well as for the communication of their status, mode and data to other systems and displays.

6.6.4 Linking recipe procedural elements and equipment procedural elements above the phase level

Clause 6.6.3 described how control recipes are linked to equipment entities when the full procedural control model is used in the control recipe. However, in some cases, it is desirable or necessary to create a master/control recipe that links to equipment procedural elements at levels higher than the phase.

EXAMPLE Examples of cases where it might be necessary or desirable to link at other levels are:

- Simplifying the creation of a master recipe by allowing it to be created using higher level recipe procedural elements

- Allowing frequently used recipe operations or recipe unit procedures to be configured as part of engineered equipment control to ensure consistency
- Allowing an entire recipe procedure that seldom changes to be embedded in equipment control
- Accommodating third-party equipment that may be provided with proprietary control systems that do not permit equipment phases to be directed by a control recipe.

If the equipment procedural element available for linkage is at a higher level than a phase in the procedural control model it must still have a unique ID for reference, be able to receive necessary formula and other associated information, interact with basic control, and be capable of carrying out the procedural functionality defined for its level.

The equipment procedural element functionality may be achieved by any method and may be structured internally in any manner. How the equipment procedural element functionality is achieved or how it is structured internally is beyond the scope of this standard.

NOTE A few possible structures or methodologies for an equipment operation include:

- It could utilize equipment phases based on the procedural control model.
- It could be composed of monolithic control code that interacts directly with equipment modules and control modules.
- It could be a written or memorized standard operating procedure followed by a person.

As a minimum a control recipe shall contain a recipe procedure. However, if the control recipe does not include all levels of the procedural model, such a recipe procedure should then be linked (by programmatic reference or by requesting some operator action) at the lowest level present in the recipe procedure hierarchy to equipment procedural elements in equipment control. Whenever a procedural element, such as a recipe procedure, recipe unit procedure, recipe operation, or recipe phase, is linked to an equipment entity, it shall be linked to an equipment procedural element of the same level. Each of the following examples illustrates a single linkage between a control recipe and an equipment procedural element.

Get more FREE standards from Standard Sharing Group and our chats

Example 1

If phases do not exist as part of the control recipe but operations do, the linking would be done at the operation level.

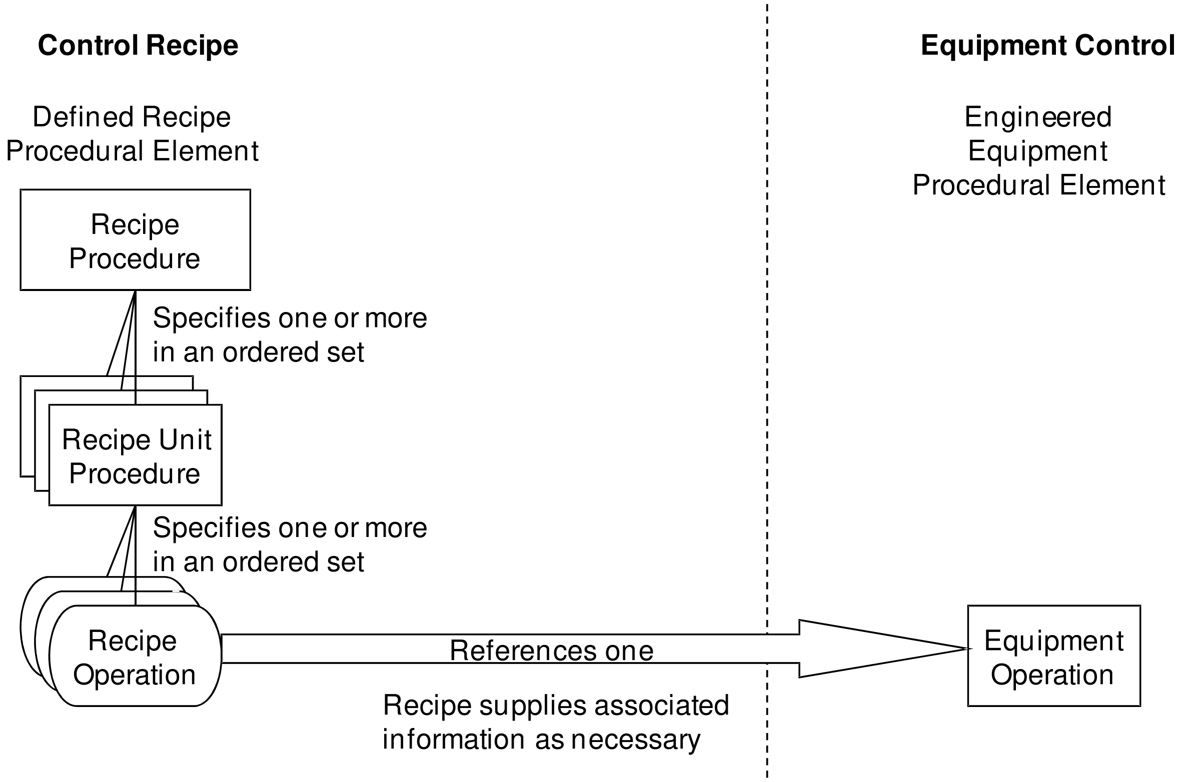

Control Recipe
Equipment Control
Defined Recipe Procedural Element
Engineered Equipment Procedural Element
Recipe Procedure
Specifies one or more in an ordered set
Recipe Unit Procedure
Specifies one or more in an ordered set
Recipe Operation
References one
Equipment Operation
Recipe supplies associated information as necessary

Figure 17 — Control recipe defined without phase level recipe procedural elements

Example 2

If neither phases nor operations exist as part of the control recipe but unit procedures do, the linking would be done at the unit procedure level.

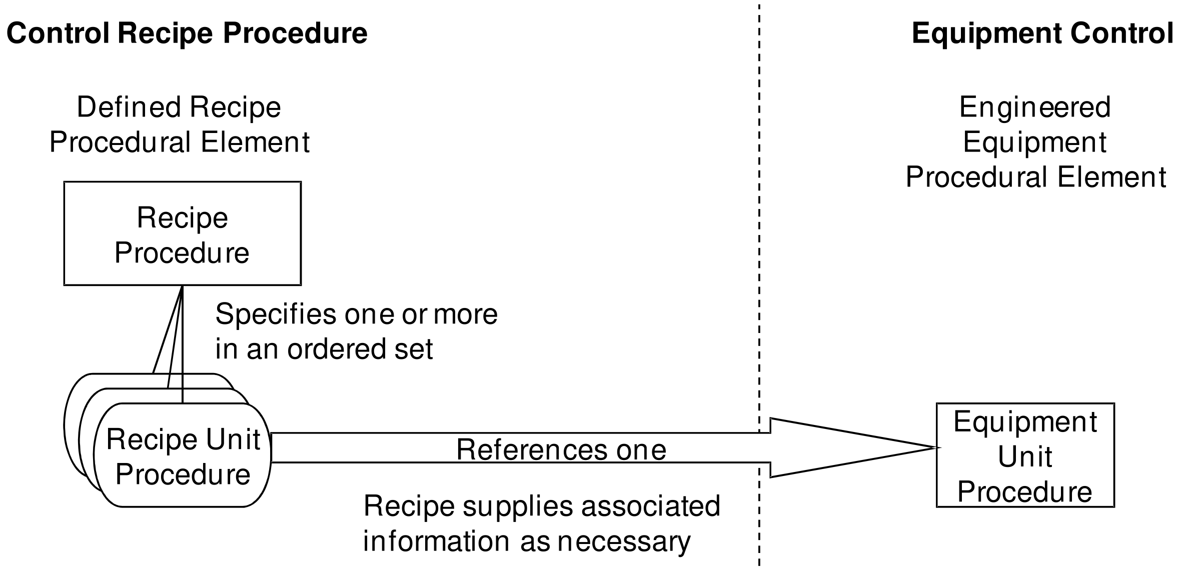

Control Recipe Procedure
Equipment Control
Defined Recipe Procedural Element
Engineered Equipment Procedural Element
Recipe Procedure
Specifies one or more in an ordered set
Recipe Unit Procedure
References one
Equipment Unit Procedure
Recipe supplies associated information as necessary

Figure 18 — Control recipe defined without operation level recipe procedural elements

Example 3

If only the procedure exists as part of the control recipe, the linking would be done at the procedure level.

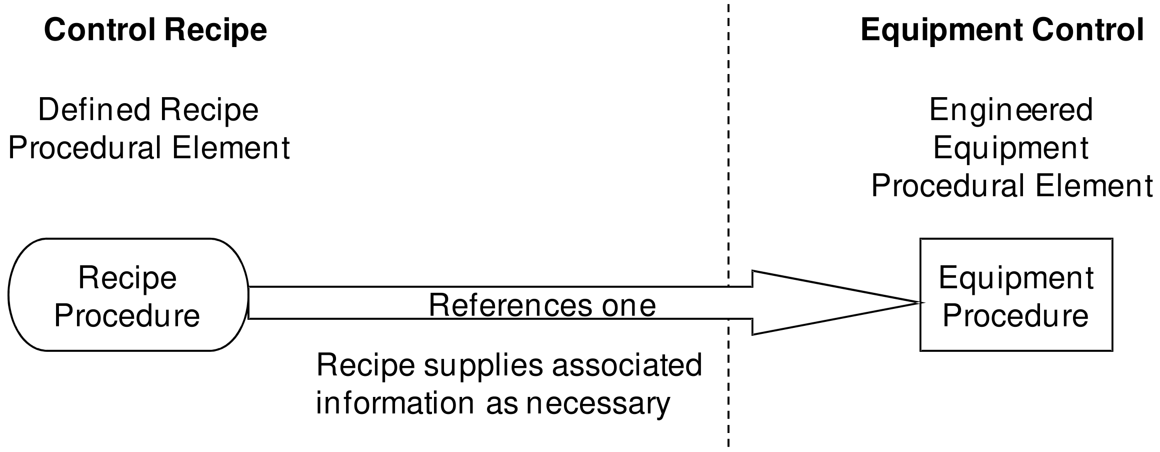

Control Recipe
Equipment Control
Defined Recipe Procedural Element
Engineered Equipment Procedural Element
Recipe Procedure
References one
Equipment Procedure
Recipe supplies associated information as necessary

Figure 19 — Control recipe defined without unit procedure level recipe procedural elements

When the full procedural model is not used in a recipe, a control recipe may contain recipe procedural elements at different levels (e.g. unit procedure, operation and phase) in the same recipe that each link to equipment procedural elements at their various levels. This is possible within the constraints of this standard and provides a significant degree of flexibility. Figure 20 provides a visual example of this. As can be seen, within the same control recipe procedure, recipe unit procedures, recipe operations and recipe phases can all reference equipment entities. Rounded rectangles have been used to help identify the recipe procedural elements that reference equipment procedural elements (EPE) in an equipment entity.

Get more FREE standards from Standard Sharing Group and our chats

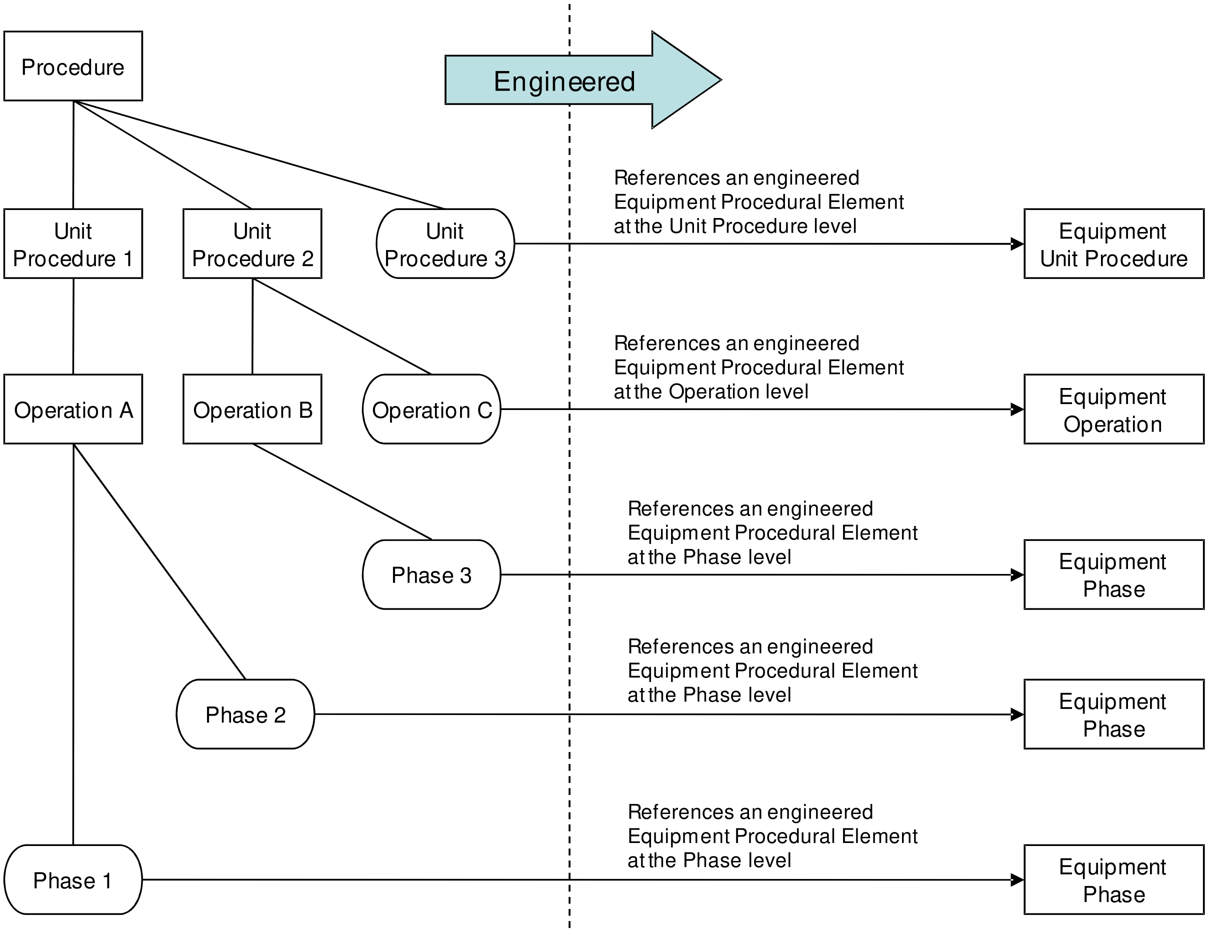

Procedure
Engineered
Unit Procedure 1
Unit Procedure 2
Unit Procedure 3
References an engineered Equipment Procedural Element at the Unit Procedure level
Equipment Unit Procedure
Operation A
Operation B
Operation C
References an engineered Equipment Procedural Element at the Operation level
Equipment Operation
Phase 3
References an engineered Equipment Procedural Element at the Phase level
Equipment Phase
Phase 2
References an engineered Equipment Procedural Element at the Phase level
Equipment Phase
Phase 1
References an engineered Equipment Procedural Element at the Phase level
Equipment Phase

Figure 20 — Referencing equipment procedural elements at different levels within the same recipe procedure

6.6.5 Control recipe procedure/equipment control collapsibility

The preceding examples illustrate how all levels of the procedural control model may be implemented with the recipe and equipment procedural element hierarchies linked at any level. As with other models of this standard, the procedural control model is collapsible. Levels in the procedural control model may be left out in a specific application. The following examples of using a collapsed procedural control model are illustrated in Figure 21 .

Example 4

If a recipe procedure addresses a single unit, the recipe procedure itself may take the place of the recipe unit procedure.

Figure 21 shows the recipe unit procedure omitted from the control recipe.

Example 5

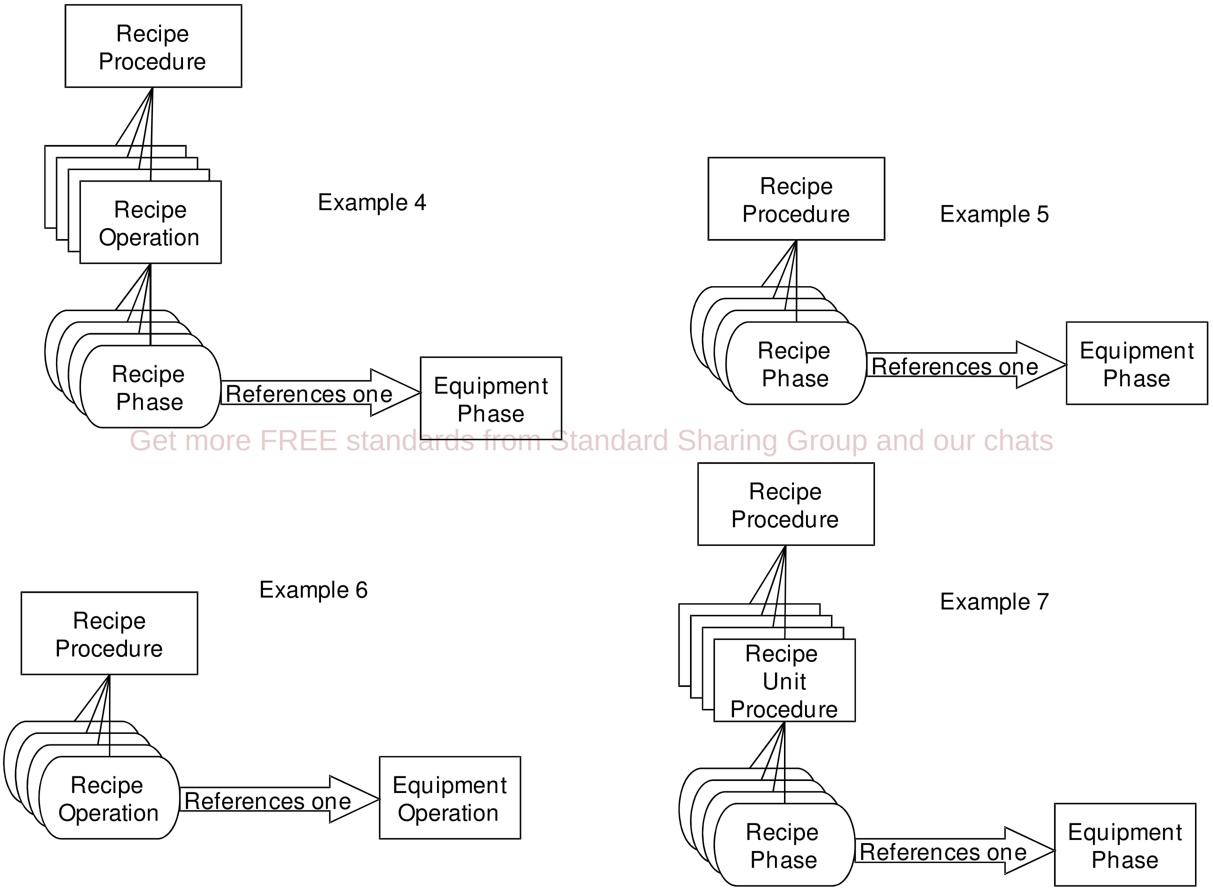
Recipe phases alone might be used to define a recipe procedure that addresses a single unit. Then the recipe procedure consists of the phases needed to accomplish the function of the procedure and the strategy needed to organize and properly sequence the phases. The recipe procedure model is collapsed to eliminate the use of recipe unit procedures and recipe operations as overtly stated subdivisions (see Figure 21).

Example 6

The phase level may be omitted if a specific application is better described with operations that are not further subdivided. Then the operation interacts directly with basic control (see Figure 21).

Example 7

The operation level may be omitted if a specific application is better described with recipe unit procedures that organize and properly sequence recipe phases directly.



Get more FREE standards from Standard Sharing Group and our chats

Figure 21 — Control recipe procedure/equipment procedure collapsibility examples

6.6.6 Linking recipe procedural elements and equipment procedural elements when using an expanded procedural control model

Clause 6.6.3 described how control recipes are connected to equipment entities when the full procedural control model is used in the control recipe. When the full procedural control model is expanded by adding new levels to the combined recipe and equipment procedural element hierarchy, the same principles used for the full and collapsed procedural models shall be followed.

7 Batch control considerations

7.1 Introduction

The batch control concepts discussed in this clause are process segmentation, exception handling, modes and states, production plans and schedules, production information, and information management.

7.2 Process and control engineering tasks

In order for required processing functions to be properly carried out in a batch manufacturing environment, the equipment structure needed, the process functionality, and the exception handling for that equipment have to be fully developed. This requires a coordinated engineering effort that continues from initial definition through the life of the batch processing facility. This clause describes the process and control engineering needed for the design of the controls needed to support the recipe hierarchy, the definition of equipment capability, and the development of the functionality required in the procedures to produce a batch.

Process and control engineering is needed at the general and site recipe levels to describe process definitions, process stages, process operations, and process actions and at the master recipe level to describe recipe procedures, recipe unit procedures, recipe operations, and recipe phases.

The precise definition of appropriate procedural elements and equipment entities is an iterative process. The dual work process is illustrated in Figure 22. Considerations affecting one decision process also affect the other. Processing considerations are the primary input to the definition (or selection) of procedural elements that will characterize functionality for associated equipment entities. Since the functionality defined will be affected by the equipment used, equipment considerations should be a secondary input. In the same way, equipment considerations form the primary input and processing considerations form the secondary input when making the definition (or selection) of equipment entities.

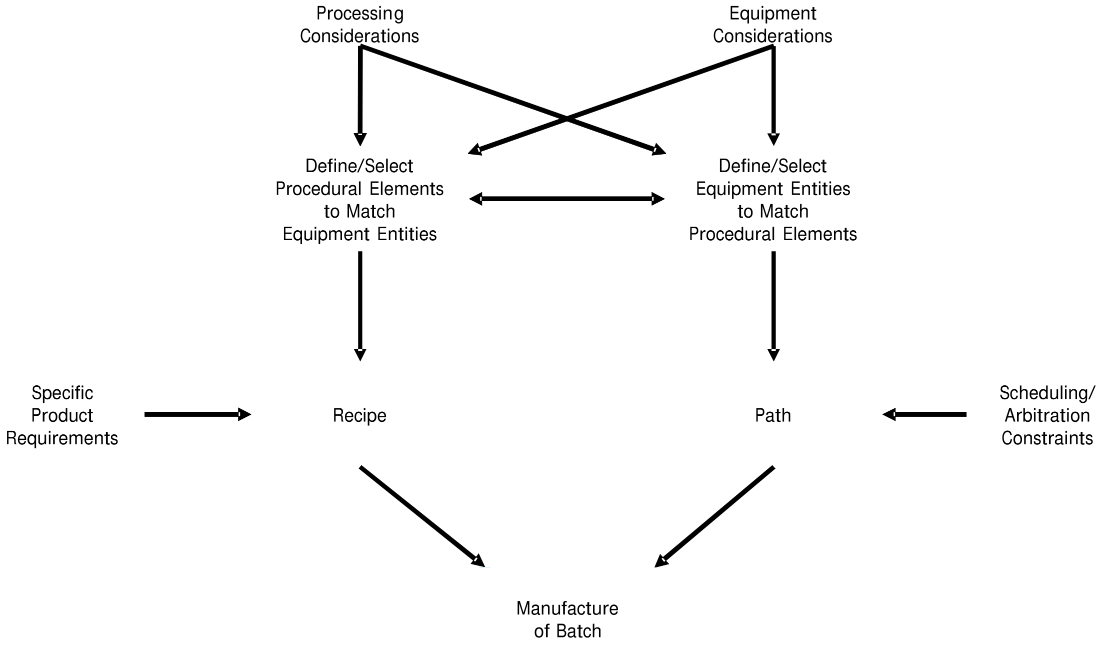

Processing Considerations
Equipment Considerations
Define/Select Procedural Elements to Match Equipment Entities
Define/Select Equipment Entities to Match Procedural Elements
Specific Product Requirements
Recipe
Path
Scheduling/ Arbitration Constraints
Manufacture of Batch

Figure 22 — Simultaneous definition/selection of procedural elements and equipment entities

Recipes can be constructed using these procedural elements and specific product information. The equipment entities are arranged into a path that is determined by scheduling and taking into account arbitration constraints. The combination of the results of these activities provides a framework within which a batch of material can be manufactured.

Get more FREE standards from Standard Sharing Group and our chats

Process and control engineering also includes the development and revision of the equipment procedural element corresponding to the recipe phases that are used to define the recipe. As far as possible, recipe and equipment procedural elements should be defined such that any reasonable functionality of a unit can be expressed in terms of these phases. They should generally not be tailored to a set of known recipes. Then, new recipes can in most cases be written by using existing recipe phases that reference existing equipment procedural element. The development and revision of recipe and equipment procedural elements is an ongoing activity that provides ongoing support to the batch manufacturing facilities. This activity is the result of the ongoing drive for continuous improvement and the periodic addition of new process technology.

7.3 Modes and states

7.3.1 Introduction

This clause discusses the modes and states of equipment entities and of procedural elements. In the preceding clauses, models describing equipment entities and procedural elements have been defined. In these models, transitions for procedural elements and for equipment entities occur within each hierarchical level. The status of equipment entities and of procedural elements may be described by their modes and states. Modes specify the manner in which these transitions take place; states specify their current status.

7.3.2 Modes

Equipment entities and procedural elements may have modes. Example modes are described in this standard in relation to batch control. The processing behavior of an equipment entity may be determined by evaluating the modes of both the associated procedural elements and the equipment entity itself.

This standard uses, as examples, three modes (automatic, semi-automatic and manual) for procedural elements, and two modes (automatic and manual) for equipment entities. Control modules contain basic control functions and will have automatic and manual modes, equipment entities running procedural control would also have a semi-automatic mode.

This standard does not preclude additional modes or require the use of the modes defined here. The functionality of the modes presented is felt to be generally useful in most batch applications. By naming the modes and including them in the standard, a defined set of terms is documented that can be used when communicating on batch control issues.

A mode determines how equipment entities and procedural elements respond to commands and how they operate. In the case of procedural elements, the mode determines the way the procedure will progress and who can affect that progression. In the case of a control module, such as an automatic block valve, that contains basic control functions, the mode determines the mechanism used to drive the valve position and who/what, such as another equipment or an operator, may manipulate it to change its state.

For procedural elements, the mode determines the way the transitions in the state model are handled. In the automatic mode, the transitions take place without interruption when the transition conditions are fulfilled. In the semi-automatic mode, manual approval to proceed is required after the transition conditions are fulfilled. Skipping or re-executing one or more procedural elements, without changing their order, is usually allowed. In the manual mode, the procedural elements and their order of execution are specified by the operator.

For equipment entities containing basic control functions, the mode determines how their states may be manipulated. In automatic mode equipment entities are manipulated by their control algorithms and in manual mode the equipment entities are manipulated by an operator.

Table 1 lists behaviours and commands associated with the example modes.

Equipment entities or procedural elements may change mode. This change can occur if the conditional logic requirements for the change are met by internal logic or by an external command such as one generated by another procedural element or by an operator. A mode change takes place only when the conditions for the change request are met.

A change of mode in one equipment entity type or procedural element type may cause corresponding changes in other types. For example, putting a unit procedure to the semi-automatic mode may cause all lower-level procedural elements in that unit to go to the semi-automatic mode, or, a safety interlock trip may cause several control modules to go to the manual mode with their outputs at minimum value. The propagation can be in either direction, from a higher level entity to a lower level entity, or conversely. This standard does not specify propagation rules.

Get more FREE standards from Standard Sharing Group and our chats

Table 1 — Example modes

Mode	Behaviour	Command
Automatic (Procedural)	The transitions within a procedure are carried out without interruption as appropriate conditions are met	Operators may pause the progression, but may not force transitions
Automatic (Basic Control)	Equipment entities are manipulated by their control algorithm.	The equipment cannot be manipulated directly by the operator.
Semi-automatic (Procedural Only)	Transitions within a procedure are carried out on manual commands as appropriate conditions are fulfilled.	Operators may pause the progression or re-direct the execution to an appropriate point. Transitions may not be forced.
Manual (Procedural)	The procedural elements within a procedure are executed in the order specified by an operator.	Operators may pause the progression or force transitions.
Manual (Basic Control)	Equipment entities are not manipulated by their control algorithm	Equipment entities may be manipulated directly by the operator.

7.3.3 States

Equipment entities and procedural elements have states that identify their current condition. The number of possible states and their names vary depending upon the requirements of the application. This standard does not require any of the example states given below or preclude additional states.

EXAMPLE 1 Examples of states for equipment entities are On, Off, Tripped, Opened, Closed, Travelling, and 35% Open.

EXAMPLE 2 Examples of states for procedural elements are Running, Holding, Paused, Stopped, Aborted, and Complete.

Equipment entities and procedural elements may transition from one state to another. Allowed state transitions may be initiated through the execution of control logic or by a valid command. The number of possible commands and their names vary depending upon the requirements of the application. This standard does not require any of the example commands given below or preclude additional commands.

EXAMPLE 3 Examples of commands for equipment entities are Start, Stop, Reset, Open, Close, and Open 35%.

EXAMPLE 4 Examples of commands for procedural elements are Start, Hold, Pause, Stop, and Abort.

EXAMPLE 5 A procedural element may transition from its Running state to its Holding state upon receiving a Hold command from an operator or another procedural element, or it may do so as a result of its internal logic.

States, commands, and transitions associated with a control recipe procedural element that contains a reference to an equipment procedural element should reflect those of the referenced equipment procedural element while they are linked.

A change of state in one equipment entity or procedural element may cause corresponding changes in other equipment entities or procedural elements. The propagation can be in either direction, such as from a higher level equipment entity or procedural element to its subordinates, or from the subordinate level to the higher level. This standard does not specify propagation rules.

EXAMPLE 6 Commanding a unit procedure to go to the Held state may cause all procedural elements in that unit to go to the Held state.

EXAMPLE 7 A safety interlock may cause all procedural elements in a unit to go to the Aborting state.

7.4 Exception handling

An event which occurs outside the normal or desired behavior of batch control is commonly called an exception. Exception handling includes the identification and evaluation of abnormal events and any manual or automatic actions initiated in response, typically affecting the modes and states of equipment entities and of procedural elements. Exception handling is an integral part of all control and in batch manufacturing generally constitutes a very large portion of the control definition.

EXAMPLE 1 Events that indicate a need for exception handling may include

- product or process problems;
- control equipment malfunction;
- hazardous conditions.

Handling of exceptions can occur at any or all levels of the equipment entity model and the procedural control model. The complete response may require a coordinated set of actions using basic, procedural, and coordination control to initially bring the process to a safe or alternate state and subsequently either terminate execution of the associated procedural elements or return them to normal operation.

Get more FREE standards from Standard Sharing Group and our chats

Exception handling is no different from desired control strategies from the standpoint that an event is detected, evaluated, and a response generated. However, exception responses generally occur outside of the desired control strategy. This difference may be characterized as follows for basic control and for procedural control:

- Typical automatic execution of basic control continuously compensates for normal deviations in the controlled variable(s) until an exception is encountered. Depending upon the potential consequences of the initiating event, exception responses may affect the modes and states of equipment entities and of procedural elements through shutdown interlocks, initiate operator interaction through alarms, and carry out further instructions from operators and from procedural control. The orchestration of actions required to resume normal operation is generally beyond the scope of basic control.
- Typical automatic execution of a procedural element includes progression through a series of normal states that each run to completion and may include conditional logic to compensate for the anticipated occurrence of deviations at defined points in its processing task. Exceptions require interruption of this normal sequence at an arbitrary point and are generally handled by initiating a transition in the affected procedural element to a state or series of states to carry out any required procedural response to the exception. For each affected procedural element, this type of transition may be initiated by an operator or automatically through conditional logic or propagation rules.

  EXAMPLE 2 An equipment breakdown or safety trip that interrupts the procedural element’s current state to execute a special event processing action is a procedural exception.

  EXAMPLE 3 A failed analytical test requiring additional processing as part of a procedural element’s normal execution is not a procedural exception.

Procedural state definitions that implement exception handling may simply modify (for example, through propagation rules) execution of the normal state that was interrupted or may specify an entirely separate action sequence. This provides a simple structure for managing exception responses associated with each concurrently active procedural element through state commands. It efficiently manages the status of the procedural element itself and any procedural actions needed both to bring the process to a safe state and, if applicable, to resume normal operation.

Different sets of exception response states may be required within each procedural element to deal with different impact severities for its defined processing task. If unique responses are required for different initiating events at the same severity level, these may be defined using different action sequences within the same state. Each set of exception responses may be initiated as a result of conditional logic, state propagation, or operator action.

The following clause (7.5 Example procedural state model) describes an example procedural state model comprising a consistent set of states, commands, and transitions with four levels of exception response, initiated by its PAUSE, HOLD, STOP, and ABORT commands. Their intended use in exception handling is as follows:

1. For exception handling, the PAUSE command could be triggered to initiate a short-term stop after allowing the RUNNING logic to continue to a safe or stable stopping point that does not require any additional shutdown or restarting actions.

   EXAMPLE 4 An equipment module commands a phase to PAUSE because a needed resource is not available, but will be available in a short time.

2. For exception handling, the HOLD command is generally applied to enable operator intervention for correction of minor process deviations from which the normal running state can be manually resumed. A HOLD could be triggered to put the procedural element and equipment into a known safe state upon encountering a minor process deviation or for a long-term stop. Exception handling can direct processing to specific states, for example sub-states of the HOLDING and HELD state to cope with the particular situation arising from certain exceptions.

   EXAMPLE 5 A batch may go into HOLD in response to a limit switch indicating a valve failure that would impact current processing.

3. For exception handling, the STOP command is usually triggered to initiate a controlled normal stop, from which the current execution of the procedural element's task cannot be returned to its normal running state. This is generally applied when manually correctable process deviations occur, but the processing task defined for the procedural element cannot continue in the usual manner.

   EXAMPLE 6 High pressure in a reactor could lead to the exception response function transferring the process to a STOPPED state, or an operator could detect some unusual condition and initiate similar action.

   EXAMPLE 7 The STOP command might also be used outside of exception handling by an operator to end a phase when a normal operation such as a titration has run to satisfactory completion but the final target condition specified in the running equipment phase has not yet been totally reached.

4. For exception handling, the ABORT command is usually triggered to initiate an immediate abnormal stop, from which the current execution of the procedural element's task cannot be returned to its normal running state. This is generally applied to initiate an emergency stop or when process deviations or equipment faults occur that cannot be corrected, requiring manual reworking, reclassification, or removal of the material being processed.

7.5 Example procedural state model

7.5.1 Introduction

The complete set of states, commands, and allowed transitions for each procedural element make up its procedural state model.

The states, commands, and transitions comprising the example procedural state model are summarized in the state transition matrix (see Table 3). The corresponding state transition diagram is shown in Figure 23. This diagram groups states together which may all respond to a common command. As illustrated, RUNNING, HELD, , PAUSING, etc. all may be commanded to STOP and a transition to the STOPPING state will be initiated.

The state transition diagram includes two general types of procedural states, defined as follows:

- A state (name ending in "ING") in which the procedural element may orchestrate a defined set of actions to complete all or part of a process-oriented task or to achieve a defined set of conditions. It implies the single or repeated execution of processing steps in a logical order, for a finite time or until a specific condition has been reached. Transition from an acting state occurs either upon completion of its defined task or upon receipt of a suitable command.
- A state for which the procedural element has previously achieved a defined set of conditions and is not permitted to direct any immediate actions. In such a state, the procedural element is maintaining a status until transitioning to an Acting state. Transition from a waiting state occurs only upon receipt of a suitable command.

The diagram also uses the terms Initial State and Final State, which refer to the normal starting and ending points for an equipment procedural element to be linked to and commanded by a recipe for execution of the process-oriented task that its recipe counterpart requires.

Get more FREE standards from Standard Sharing Group and our chats

NOTE The use here of the term Initial State is meant specifically in relation to its interaction with recipes, not as the state that the equipment procedural element is expected to be in upon power-up of the controlled equipment or of the control system itself, which for example might alternatively be designated for any particular instance as one of the Final States.

This model can be applied to a wide range of situations and is given as a simple illustration of procedural states. An example of a more expansive model is found in Annex D.

7.5.2 Procedural states

For the example procedural state model, the list of valid procedural states are given in Table 2.

Table 2 — State descriptions in the example procedural state model

State	Description
IDLE (Initial State)	Waits for the procedural element to be allocated and started, both of which may be initiated through execution of another procedural element or by other means. Upon receiving a START command while the associated permissives are satisfied, control passes to RUNNING.
RUNNING	Directs normal sequencing of the process-oriented task and handling of routine variations that can be addressed without interruption by exception handling. Upon completion, control passes to COMPLETE.
COMPLETE (Final State)	Waits in the final state for a RESET command after the process-oriented task has run to completion. Upon receiving a RESET command, control passes to IDLE.
PAUSING	Temporarily interrupts the RUNNING state after receiving a PAUSE command. While in the PAUSING state, the normal RUNNING logic continues to execute until reaching a defined stopping point. This could be initiated to halt execution of the normal RUNNING logic when its processing actions have reached a safe or stable stopping point, such as for a short-term stop that does not require any additional shutdown or restarting actions. Upon reaching a defined stopping point in the normal RUNNING logic, the RUNNING logic is halted and control passes to PAUSED.
PAUSED	Waits for a RESUME command before returning to RUNNING. Upon receiving a RESUME command, control passes to RUNNING and resumes normal execution immediately following the stopping point that was reached while PAUSING.
HOLDING	Temporarily interrupts RUNNING, PAUSING, PAUSED, or RESTARTING after receiving a HOLD command. This could be initiated to put the procedural element and equipment into a known safe state, after which the normal running state can be manually resumed. Upon completion, control passes to HELD.
HELD	Waits for a RESTART command before proceeding to the appropriate restarting state. Upon receiving a RESTART command while the associated permissives are satisfied, control passes to RESTARTING.

State	Description
RESTARTING	Performs any restarting actions that must be executed before returning to RUNNING after being HELD. Upon completion, control passes to RUNNING.
STOPPING	Terminates RUNNING, PAUSING, PAUSED, HOLDING, HELD, or RESTARTING after receiving a STOP command. This is usually initiated to perform a controlled normal stop, after which the current execution of the process-oriented task cannot be returned to the RUNNING state. Upon completion, control passes to STOPPED.
STOPPED (Final State)	Waits in the final state for a RESET command after the process-oriented task has been stopped. Upon receiving an RESET command, control passes to IDLE.
ABORTING	Abnormally terminates RUNNING, PAUSING, PAUSED, HOLDING, HELD, RESTARTING, STOPPING, or STOPPED after receiving an ABORT command. This is usually initiated to perform an immediate abnormal stop, after which the current execution of the process-oriented task cannot be returned to the RUNNING state. Upon completion, control passes to ABORTED.
ABORTED (Final State)	Waits in the final state for a RESET command after the process-oriented task has been aborted. Upon receiving a RESET command, control passes to IDLE.

Get more FREE standards from Standard Sharing Group and our chats

7.5.3 Procedural commands

For this example procedural state model, the list of valid commands are the following:

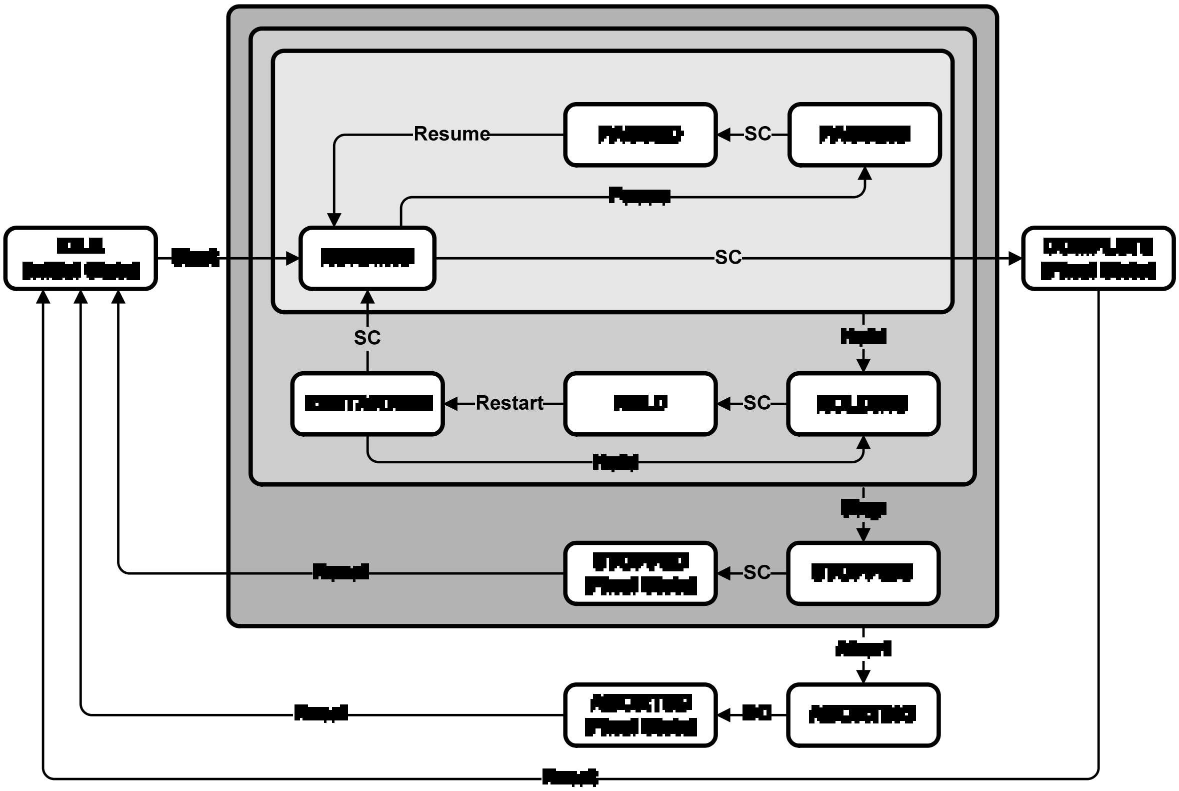
- START: This command orders the procedural element to transition from the IDLE state to the RUNNING state.
- RESET: This command orders the procedural element to transition from the COMPLETE, STOPPED, or ABORTED state to the IDLE state.
- PAUSE: This command orders the procedural element to transition from the RUNNING state to the PAUSING state.
- RESUME: This command orders a procedural element to transition from the PAUSED state to the RUNNING state.
- HOLD: This command orders the procedural element to transition from the RUNNING, PAUSING, PAUSED, or RESTARTING state to the HOLDING state.
- RESTART: This command orders the procedural element to transition from the HELD state to the RESTARTING state.
- STOP: This command orders the procedural element to transition from the IDLE, RUNNING, PAUSING, PAUSED, HOLDING, HELD, or RESTARTING state to the STOPPING state.

- ABORT: This command orders the procedural element to transition from the IDLE, RUNNING, PAUSING, PAUSED, HOLDING, HELD, RESTARTING, STOPPING, or STOPPED state to the ABORTING state.

Table 3 — State transition matrix for example states for procedural elements

Current State	**Transition End State when Sequence Complete**	**Transition End State upon receiving each Valid State Command**							
		START	**STOP**	**HOLD**	**RESTART**	**ABORT**	**RESET**	**PAUSE**	**RESUME**
IDLE		RUNNING							
RUNNING	COMPLETE		STOPPING	HOLDING		ABORTING		PAUSING	
COMPLETE							IDLE		
PAUSING	PAUSED		STOPPING	HOLDING		ABORTING			
PAUSED			STOPPING	HOLDING		ABORTING			RUNNING
HOLDING	HELD		STOPPING			ABORTING			
HELD			STOPPING		RESTARTING	ABORTING			
RESTARTING	RUNNING		STOPPING	HOLDING		ABORTING			
STOPPING	STOPPED					ABORTING			
STOPPED						ABORTING	IDLE		
ABORTING	ABORTED								
ABORTED							IDLE		

NOTE The states ending with “ING” are acting states. If their logic completes normally, then a state transition to the state listed under “Transition End State when Sequence Complete” occurs. For example, if the RUNNING state completes normally, then the state automatically transitions to COMPLETE. Execution of the acting states (ending in -ING) is governed by the mode.



Notes: 1. SC = State Change as a result of state actions completed.
2. Actions of an equipment procedural element are generally defined by its Active States.
3. The light, light-medium, and light-medium +dark grey boxes represent collections of states that can be preempted using the Hold, Stop, and Abort commands, respectively.

Get more FREE standards from Standard Sharing Group and our chats

Figure 23 — State transition diagram for example states for procedural elements

7.6 Batch schedules

7.6.1 Introduction

Production plans and schedules define the production requirements for the enterprise, sites, areas, and process cells. Since these levels of the physical model operate on different time horizons, a number of different types of plans and schedules are typically needed within an enterprise. The ANSI/ISA-95 and IEC 62264-1 standards address this topic in a much broader context for all types of manufacturing. A detailed discussion of those standards or the various types of plans and schedules is outside the scope of this standard. Only the scheduling needs for batch manufacturing at the process cell level, the batch schedule, are discussed.

Batch schedules identify the batches to be produced, how much to produce, what recipes to use, and from what sets of equipment the actual production units may be selected.

The ISA-88 batch schedule corresponds to the ISA-95 dispatch list, but is constrained to the process cell and does not directly deal with resources other than equipment.

The batch schedule typically contains more detailed information than production plans and schedules aimed at higher levels in the enterprise. It contains information such as the batches that are to be produced, the size of the batch, and when they are required for a specific process cell. It identifies which batches are to be made, their order, and the equipment to be used. This schedule also deals with issues such as personnel requirements, raw material options, and packaging requirements.

Time horizons for the batch schedule are dependent on the speed of the processes and might be measured in minutes, hours, shifts, or days. The batch schedule is based on the specific resources and requirements of the process cell. The possible paths and equipment options may be determined when the batch schedule is generated. For the batch schedule to be totally meaningful, the schedule may need to be redone any time there is significant variance from the time projections, resource assumptions, or other anticipated factors on which the schedule was based. For example, the schedule may have to be updated if a task is not completed close to the scheduled time. Whether that task is delayed or whether it is completed ahead of time, the primary concern is whether that task can affect other schedules in this process cell or other associated process cells.

The following is the typical information that may be found in a batch schedule (if pre-assigned):

- Produced material name
- Master recipe name
- Quantity (with engineering units) of product
- Equipment and materials permitted to be used, such as path and raw material
- Order of initiation and priority
- Lot ID
- Batch ID
- Projected start time and end time
- Disposition of the finished batch
- Specific customer requirements

A key to efficient batch manufacturing is a comprehensive business process that links the various plans and schedules with batch data collection. Batch data collection is the source of timely information that provides feedback so that these plans and schedules can be fine tuned. During the actual manufacturing of a batch, information is often needed in real time so that schedules can be updated within a short time horizon. This update information also allows the user to be kept apprised of the status of lots and/or batches in the schedule.

7.6.2 Campaign

A series of batches can be called a campaign. The term describes a common method of operation of production facilities that relate batches together. Campaigns may be defined for one or more reasons.

EXAMPLE Examples for campaign definition include:

- Each batch is smaller than the required amount of material to be produced, so that multiple batches are required to meet production requests. The batches are all tied to a specific production request and can collectively be described as a product campaign.

- Multiple batches may be produced using the same master recipe, even if the production requests are not related, except that they specify the same master recipe. This can be described as a batch campaign.

- It is preferred for operators to initiate a campaign to run multiple batches rather than starting each batch individually.

- It allows the data from multiple batches in a campaign to be combined and analysed easily in reports.

- A series of recipes, both production and non-production (e.g. cleaning), may be related in a sequence of execution to form a campaign.
- A series of recipes may be executed to perform non product related procedures, such as cleaning or sterilizing of equipment. This could be described as cleaning or sterilization campaigns that execute the same non-product master recipe against different pieces of equipment.

Campaigns may not be limited to the process cell. They are the result of a planning process that results in a series of planned batches for one or more process cells. In this case, the batches within a single process cell may not appear as a campaign.

7.7 Production information

7.7.1 Introduction

This clause discusses information that is generated in the course of production. Information needs to be collected and made available to various levels of the enterprise. The type of information needed varies between different parts of the enterprise. At the enterprise level, for example, summary information may be all that is needed. Examples include the amount of production of a particular product that was achieved at a specific site or at all sites, or how much product is available in inventory.

Process development may need detailed processing information on the individual batches in order to perform statistics and comparisons. At the process cell level where the batches are actually executed, there is a need for more detailed information in order to monitor the day-to-day production, to perform adjustments to the schedule, or to adjust processing from batch to batch.

The ANSI/ISA-95 and IEC 62264-1 standards address this same topic in a much broader context for multiple levels in the enterprise and for all types of manufacturing. A detailed discussion of those standards or the various types of information or methods required is outside the scope of this standard. Only a subset of the needs for batch manufacturing at the process cell level is defined in this part.

Get more FREE standards from Standard Sharing Group and our chats

Production information may be batch specific or it may be common to several or all of the batches produced. See Part 4 of this standard for additional information on production information.

7.7.2 Batch-specific information

The batch-specific information may include the following:

- A copy of the control recipe that was used to make the batch. This may not be identical to the original recipe because of operator changes, equipment problems, etc. It may be desirable to record both the original recipe and the actual recipe.
- Recipe data. This is actual process data that corresponds exactly to the recipe formula, such as the amount and type of material charged. This can then be compared to the original recipe.
- Recipe-specified data. This is data whose collection is specified by the recipe. An example is process control information to be trended.
- Summary batch data. This is data such as utilities consumption, equipment run times, and temperatures for the entire batch.
- Operator comments
- Continuous data. This is process data that is collected independent of specific events within the batch with the purpose of giving an accurate history of that measurement.

- Event data. This is data from predictable and unpredictable events, such as recording start and stop times of procedural elements, or unpredictable process or equipment events.
- Operator data. This includes any operator intervention that may affect the processing of the batch (includes operator's ID).
- Analysis data. This is data that is related to off-line measurements or analyses such as measured variables, operator ID, lab technician ID, time of entry of results, and time of sample.

7.7.3 Common (non-batch specific) batch information

EXAMPLE Some examples of common (non-batch specific) batch information include:

- Quality control information. This is information related to monitoring raw material qualities and processing quality.
- Utility systems information. This is process information for equipment such as process heating and cooling that do not produce batches themselves but support equipment that does produce batches.
- Equipment history. This is historical information, such as equipment utilization, calibration, and maintenance.
- Operational documentation. This includes documentation such as production volumes, material consumption summaries, and inventory statistics.
- Materials information. This is typically information such as quality information and packaging and labelling information of input and output materials.

7.7.4 Batch history

All recorded information pertaining to a batch is referred to as the batch history. The batch history will typically include the batch-specific information. Common (non-batch specific) batch information may be included in the batch history. Since information of this nature typically applies to all or several batches being processed in a process cell, it may be included in the individual batch histories by reference.

In many regulated industries, the record of the batch history is as important as the product itself. Without reliable and accurate batch record keeping, product quality and traceability cannot be ensured. Complete batch record keeping also provides information that is invaluable in process analysis and continuous improvement efforts. See Part 4 of this standard for additional information on batch history and representations of batch production records.

Batch history should be stored in a way that makes it possible to associate the data with that batch (or batches) to which it relates and the processing that has taken place. This means that, in addition to the specific batch identity, the data should be associated with the actual execution of the appropriate procedural elements, where relevant. The structure of the executed procedure may differ from what is specified in the original recipe because of operator intervention, exception handling, or even planned diversity in the procedure, such as changes caused by varying resource limitations.

7.7.5 Batch reports

The extraction of data related to one or more batches is called a batch report. The extraction and ordering of the data in a report may vary based on the intended recipient of the batch report. Some of the typical recipients of batch reports and the types of information typically included in their reports are

- Production management: These batch reports typically provide key economic information on the processing result and resource utilization from multiple batches.

- Product development: These batch reports typically include detailed process information for an individual batch or compare similar data between a group of batches.
- Plant operations: These batch reports typically include the data collected to the current point of processing.
- Quality management: These batch reports typically contain information for documenting batch quality, which may be useful in quality statistics.
- Authorities: These batch reports are typically provided as documentation of production complying with regulations.
- Customers: These batch reports usually are documentation of product quality and process uniformity.

Get more FREE standards from Standard Sharing Group and our chats

8 Activities and functions in batch control

8.1 Introduction

This clause discusses the major activities of batch control and the supporting functions.

8.2 Control activity model

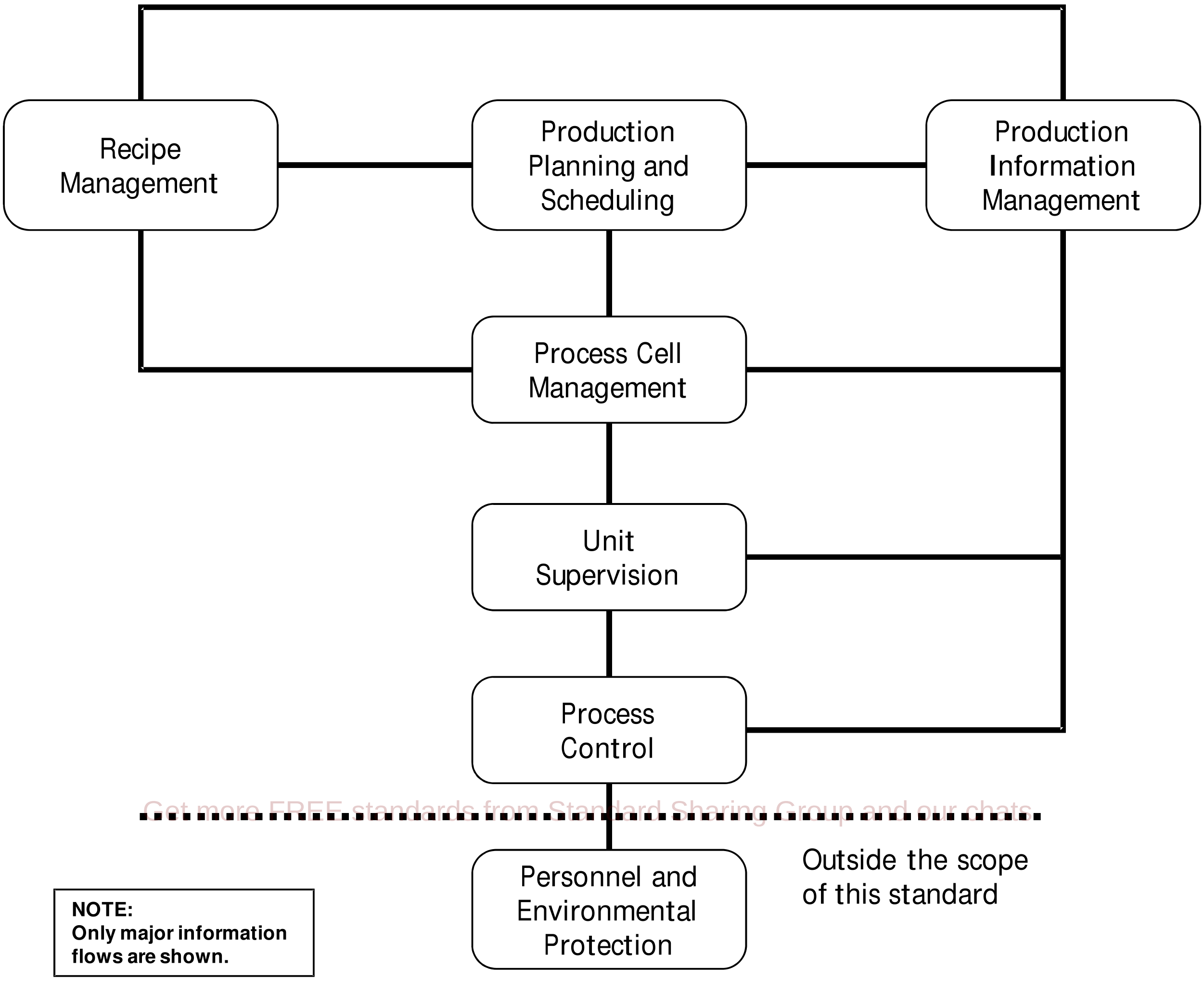
8.2.1 Introduction

Seven control activities are defined as represented in Figure 24. Each activity is modelled to show the functions within them. These functions define how batch processes are controlled using equipment and/or personnel.

The Control Activity Model in Figure 24 provides an overall perspective of batch control and shows the main relationships between the various control activities. It is not intended to show all relationships. The relationships illustrated are achieved via information flow between the control activities. The purpose of this drawing is simply to show where there is a relationship and not to define that relationship. The major relationships are defined later in this clause. Some of the relationships shown in Figure 24 are not discussed further in this standard.

The control activities relate to the following requirements in a batch manufacturing environment:

- The need to have functions that can manage general, site, and master recipes implies a need for the Recipe Management control activity.
- Production of batches should occur within a time domain that is planned and subsequently carried out. Production Planning and Scheduling is the control activity where these functions are discussed.
- Various types of production information should be available, and the collection and storage of batch history is a necessity. The Production Information Management control activity in the model covers these functions.
- Control recipes should be generated, batches should be initiated and supervised, unit activities require coordination, and logs and reports should be generated. These functions fall under the Process Cell Management control activity in the model.
- There are many functions needed at the Unit Supervision control activity level. For example, there is a need to allocate resources, to supervise the execution of procedural elements, and to coordinate activities taking place at the Process Control level.
- In Process Control, functions are discussed that deal directly with equipment actions such as the need to implement functions using regulating equipment and/or state-oriented equipment.


Recipe Management
Production Planning and Scheduling
Production Information Management
Process Cell Management
Unit Supervision
Process Control
Personnel and Environmental Protection
Outside the scope of this standard
NOTE:
Only major information flows are shown.

Get more FREE standards from Standard Sharing Group and our chats

Figure 24 — Control activity model

Finally, the safety of personnel and the surrounding communities should be a prime concern, along with protection of the environment. The Personnel and Environmental Protection control activity covers these functions.

8.2.2 Information handling

8.2.2.1 Introduction

One dimension of the control activity model is its description of information flow throughout the levels. As such, there are a number of information handling functions that can be applied to all categories of data addressed by the control activity model. These are applicable regardless of the combination of manual and computerized systems that are established at a site. Additional information handling aspects that are specific to a particular control activity are described within their respective clauses.

8.2.2.2 Reference information

The batch manufacturing enterprise may incorporate activities that fall outside the scope of this standard.

EXAMPLE Examples of activities outside of the scope of this standard include:

- material inventory management
- process and product development
- customer service support
- regulatory reporting and process validation
- inter-departmental coordination, such as production versus support services

To provide an interface to these information sources, the control activities discussed in this clause shall store information in a way that provides a usable, accessible data source to these external activities. Similarly, each control activity should have the ability to access relevant reference information as needed to fulfill its function.

EXAMPLE Examples of reference information include:

- sales or marketing data, including customer orders or other statements of product demand
- raw material vendor data
- final products specifications
- costing data
- research and development data
- standard consumptions of raw materials and standard yields for the products manufactured
- rate information for the various process cells
- equipment capability specifications
- operational procedures for equipment maintenance and process safety
- human resource information
- quality control information such as the procedure used to perform a particular laboratory analysis
- regulatory requirements

Reference information may be enterprise-wide, site-wide, area-wide, or process cell-wide.

8.2.2.3 Security

Within the control environment, information is used to impact the functions, to communicate between levels and entities, and to provide communication to functions outside of the control activity model. Access to this information should be to ensure that only authorized and/or qualified resources can affect the information.

8.2.2.4 Availability

Control activity information should be stored and retrieved in a way that provides the necessary safeguards to ensure access to critical data. The time necessary to recover access to the data in case of loss at one location should be considered carefully. These considerations will vary based on the different levels of the control activity model, the types of information, and the level of detail required.

8.2.2.5 Archival

Removal of information from the control activity and into a long-term archive is often desirable to improve storage efficiency and recoverability. Once archived, it should be possible to retrieve the archived data in a usable form. For example, once a master recipe is no longer in active use, it would be useful to be able to extract all information (both structural and historical) related to that master recipe from the main repository.

8.2.2.6 Change management

Information that defines control — including configuration of equipment control and recipes — should be subject to formal change management. Means should be provided to support

- requests for and authorization of changes
- version numbering and documentation
- validation of changes
- audit tracking

Change management should also include restrictions and checks necessary to maintain the integrity of the configuration.

EXAMPLE It may be necessary to prevent a recipe creator from modifying a procedural element in use by an active recipe.

8.2.2.7 Reference tracking

Tracking of information references — for example, which definitions are used within which others or which served as the basis for others — is important in analysis of production performance and in demonstrating compliance with production guidelines. This function should provide a means to attach written comments about the changes, to assist in subsequent interpretation.

Get more FREE standards from Standard Sharing Group and our chats

8.3 Recipe management

8.3.1 Introduction

Recipe Management is made up of the functions that create, store, and maintain general, site, and master recipes. The overall output of this control activity is a copy of the master recipe that is made available to Process Cell Management, which uses it to create a control recipe.

Recipe Management will be discussed in terms of managing the three levels of recipes and defining the procedural elements used in the recipe procedures (see Figure 25).

NOTE The ANSI/ISA-95 and IEC 62264-1 standards address this same topic in a much less batch manufacturing specific way for all types of manufacturing under the general heading of Product Definition. The remainder of this clause deals with the specific requirements for batch manufacturing.

8.3.2 Manage general recipes

Manage general recipes is the function by which general recipes are created, maintained and stored. The specific processing requirements furnished by the process development activity for the product being considered serve as the basis for the general recipe.

In connection with the definition of the individual general recipe, the following capabilities may be required:

- Selecting and combining procedural elements to create a general recipe process definition

- Incorporating formula information
- Specifying equipment requirements and other information
- Maintaining the general recipe
- Managing changes to general recipes

8.3.3 Define general recipe procedural elements

The *define general recipe procedural elements* function creates, maintains and makes available for subsequent use, the procedural elements that are used as building blocks in general recipe and site recipe process definitions. See Part 3 of this standard.

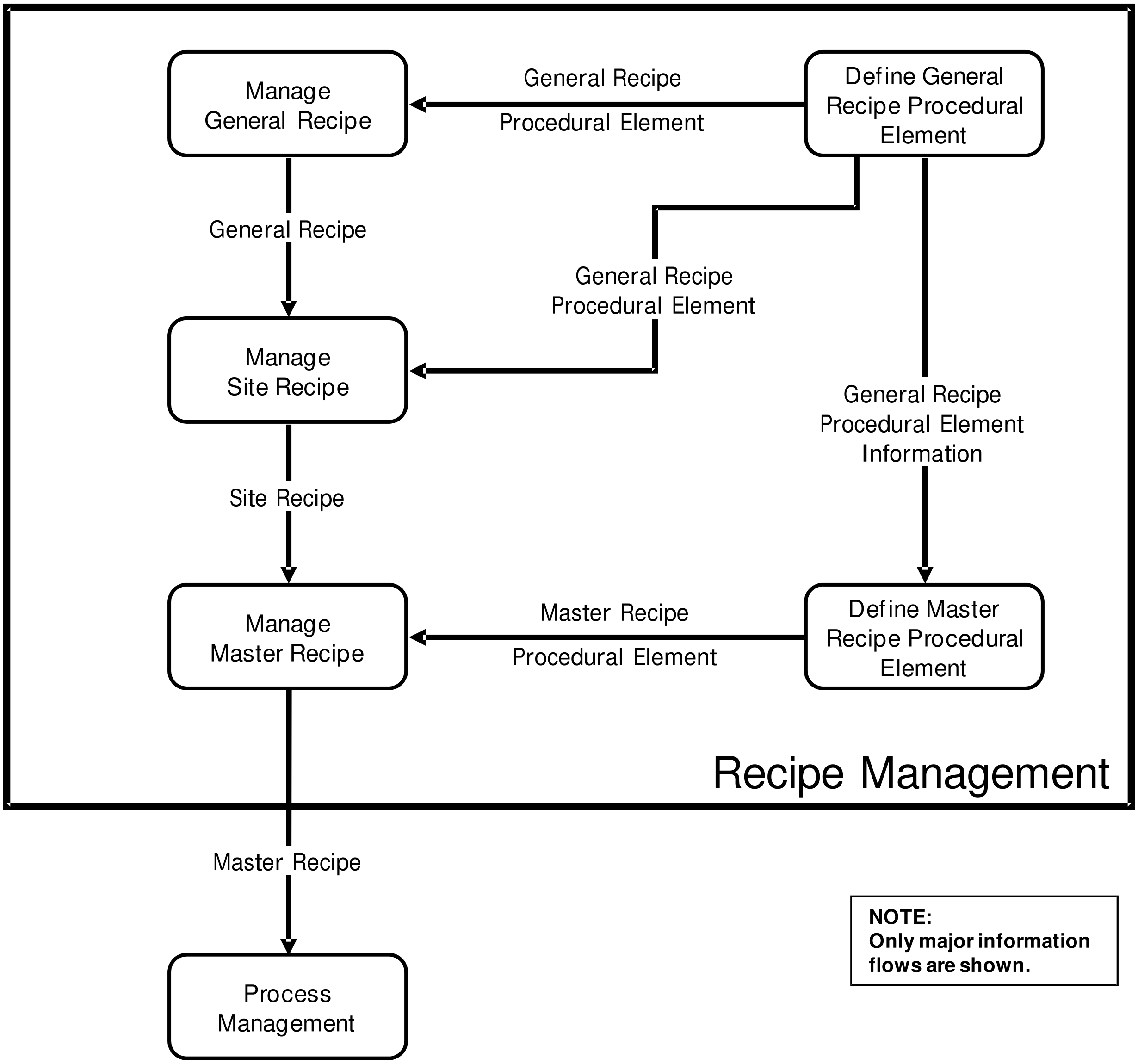


Figure 25 — Recipe management

General recipe procedural element information should be made available to define the master recipe procedural elements function. In this way, the process intent of the general recipe procedural elements may be known at the master recipe level.

In connection with the definition of the individual general recipe procedural elements, the following capabilities may be required:

- Naming the individual general recipe procedural elements
- Specifying associated formula data
- Describing the intended processing functionality
- Combining lower level procedural elements and specifying the sequence of execution
- Creating, modifying, and archiving general recipe procedural elements
- Maintaining an inventory of procedural elements available
- Managing changes to procedural elements

8.3.4 Manage site recipes

Manage site recipes is the function by which site recipes are created, maintained and stored. A site recipe is created by combining the information of the appropriate general recipe with site specific information, or creating a site recipe from production requirements. If additional or alternate procedural elements are required, only those defined under the *define general recipe procedural elements* function are used.

8.3.5 Manage master recipes

Manage master recipes is the function by which master recipes are created, maintained and stored. Master recipes are defined based on the specific processing requirements for the batch in question. These specific processing requirements may be expressed in a general or site recipe.

Get more FREE standards from Standard Sharing Group and our chats

The transformation of the site recipe into a master recipe may be a complex task. The creation of a procedure, based on predefined procedural elements, should match the intent of the site recipe process definition. Transformation (or creation) of the content of the formula follows the same general logic that is used to map process actions to recipe phases. The batch size is fixed, or the range of batch sizes permissible for the recipe is established, if there are constraints on the degree of scalability. Formula information is adjusted accordingly. The equipment requirements are transformed into requirements that can be verified against the actual target equipment. See Part 3 of this standard for additional information.

In connection with the definition of the individual master recipe, the following capabilities may be required:

- Selecting and combining procedural elements to create a master recipe procedure
- Incorporating formula information
- Specifying equipment requirements and other information
- Creating, modifying, and archiving master recipes and maintaining the recipe headers
- Maintaining an inventory of master recipes
- Managing changes to master recipes

8.3.6 Define master recipe procedural elements

The *define master recipe procedural elements* function creates, maintains and makes available for subsequent use, the procedural elements used in master recipe procedures. These become the building blocks of the master recipe procedure.

The master recipe procedural elements should reflect the processing capabilities required by master recipes. If these are generated from general and site recipes, then process stages, process operations, and process actions will map into unit procedures, operations, and phases. This function defines the relationship between process actions and phases, between process operations and operations, and between process stages and unit procedures. It also defines the general scope of procedures, unit procedures, operations, and phases to allow maximum consistent use of pre-defined procedural elements across the range of products to be made in the facility.

The master recipe procedural elements should, at least at the recipe phase level, be able to identify equipment procedural elements that will be linked when the derived control recipe is executed. A close coordination with the engineering of the equipment procedural elements should therefore take place, ensuring that the recipe procedural elements adequately reflect the control capabilities of the target equipment. If required, any new functionality is made available through creation of new procedural elements, along with associated control and equipment modifications (see Clause 7.2).

In addition to providing the building blocks for the master recipe procedure, this function may also define constraints on the configuration of master recipes, such as rules on the allowable order of recipe phases and limitations in the recipe creator's right to use recipe phases as building blocks. The determination of such constraints is made based on many factors, such as safety, complexity of the recipe creator's task, required flexibility, and validation of individual procedural elements.

In connection with the definition of the individual procedural elements, the following capabilities may be required:

- Naming of the individual procedural elements
- Specifying parameter variables
- Describing the intended processing functionality
- Combining lower level procedural elements and specification of the sequence of execution
- Creating, modifying, and archiving master recipe procedural elements
- Maintaining an inventory of procedural elements available
- Managing changes to procedural elements

8.4 Production planning and scheduling

Production Planning and Scheduling is a high level control activity on a peer level with Recipe Management and Production Information Management.

NOTE The ANSI/ISA-95 and IEC 62264-1 standards address the topic of production planning and scheduling for multiple types of manufacturing. Consideration of those standards is recommended. This clause is limited to production planning and scheduling specific to batch manufacturing to provide an overview of this activity in the control activity model.

It is the decision process associated with producing a batch schedule that is provided to Process Cell Management. Although several functions would need to be collected together to make up this control activity, most of those functions are outside the scope of this standard. This clause will consider only one of these functions: *Develop batch schedules*.

The *develop batch schedules* function accepts inputs from sources such as other types of schedules, master recipes, and resource databases, and, based upon a scheduling algorithm (automated or manual), develops a batch schedule (see Clause 7.6 for a list of typical information in a batch schedule).

The following capability is typically included in this function:

- Developing a batch schedule based on information from the appropriate source and some scheduling algorithm
- Developing a revised batch schedule on demand based on significant changes in batch progress and process cell status information provided by Process Cell Management
- Allowing for manual intervention into the scheduling process
- Determining the availability of resources as an input into the scheduling process
- Providing a procedure or method for batch sizing along with a means to organize the production of batches
- Determining the feasibility of the schedule based on the target equipment and available ingredients.

8.5 Production information management

8.5.1 Introduction

Production Information Management is a high level control activity on a peer level with Recipe Management and Production Planning and Scheduling. It is the control activity that is involved in collecting, storing, processing, and reporting production information.

The non-batch-related use of production information is not dealt with in this clause, but in actual applications the management of batch-related information and non-batch-related information may very well be integral. Both batch-related and non-batch-related information may be used as input to higher-level functions such as the generation of production reports to management. These activities will not be modelled in this part of the standard. See Part 4 of this standard for further information on batch production records.

Get more FREE standards from Standard Sharing Group and our chats

NOTE The ANSI/ISA-95 and IEC 62264-1 standards address the entire topic of information management for multiple types of manufacturing. Part 4 of this standard is focused on specific aspects of batch information. Consideration of those standards is recommended. This clause is limited to production information management specific to batch manufacturing.

Although several functions would need to be collected together to make up the entire Production Information Management control activity, most of those functions are outside the scope of this standard. This clause will consider only one of these functions: *Manage batch history*.

Batch history is a collection of data related to one batch. It may be organized in one or more files or tables per batch, or it may be present as a part of a database and retrievable via key fields, etc.

Batch history is built up of entries. An entry is a portion of information on the batch representing data about one event logged into the batch history in one action.

Manage batch history is the function that typically includes the following capabilities:

- Receiving and storing information from other parts of the overall batch control application on batches
- Manipulating historical data
- Producing batch reports

The Manage batch history function is performed regardless of the equipment used or when a batch is produced. For example, lab data often may be added after the execution of the batch.

8.5.2 Receiving and storing batch history information

8.5.2.1 Introduction

The entering of data from the outside into batch history is initiated from Process Cell Management, Unit Supervision, and Process Control.

8.5.2.2 General collection and storage guidelines

All of the data for the batch history should be collected and stored in a way that includes or gives simple access to

- batch identification;
- absolute time stamp (real time);
- identification of procedural elements with which the data is associated;
- time relative to the start or end of a batch or of the execution of a procedural element;
- product data;
- equipment utilized
- operator comments.

Adequate storage capacity is needed for the required number of batch histories. This should include sufficient capacity to store the batch histories of all running batches, and for finalized batches until appropriate actions have been taken (reports printed, long term backup or whatever action is specified).

To the extent that the storage time requirement exceeds the storage capacity of *manage batch history*, the capability should exist to export the batch histories onto long-term storage media or external systems. It should be possible to retrieve these batch histories for further extraction of data.

Reports or displays about the batch archive may include information such as number of batches in the archive, amount of data, status [finalized, printed, archived in long term archive, etc.].

8.5.2.3 Reliability of batch history entries

The requirements for reliability will vary from application to application and between the different entry types. In the following, a number of issues of reliability are described. For each type of entry, the appropriate level of reliability should be selected to match the needs of the individual application. Reliability issues include

- access control: control of access to the data-gathering system, including the configuration and the actual data collected;
- audit trail: identification of all manipulation that happened with each individual piece of information — including identification of the person or controls involved, the time and, in some cases, an explanation;
- logging reliability: specification of the required reliability of logging. Three levels may be distinguished:
  - Nice to have — no specific action in case of failure. Examples include data for optimization, equipment reliability statistics, etc.;
  - Limited holes acceptable if the failure is indicated in the batch history (logging absent from . . . to . . .);

  - Critical — data should be available. If it is missing, then backup procedures should be possible (electronic or manual backup, possibility of reconstruction, etc.).

  The importance of exact logging of the latter type of information may be equivalent to the achieved product quality, either for financial reasons (accounting) or for product safety/responsibility reasons. Therefore the receiving function should be capable of providing feedback information on the general status of the receiving function (as well as specific confirmation feedback for each entry to the control activity that performs the logging) enabling them to perform buffering, redundancy or reintegration activities or, if required and allowed, to hold up the process.

- completeness of history: The level of detail in information collected should be well defined in the recipe, or equipment. It should be possible to see if there are omissions from anticipated tasks. This may be due to a task not occurring or, that the task occurred but the level of detail in records did not capture that task.
- logging of actual historic information: Batch history entries should, to the largest possible extent, reflect the actual physical/chemical events that influence the batch, not only what was anticipated in the recipe. That means that the character and amount of data logged will vary due to the variations in batch production.
- long-term consistency: The extent to which the interpretation of batch data relies on information outside of the batch history, such as cross reference lists between actual tags and batch entry tags or names of variables, should be well described. Such information should be stable in the long term. If changes or modifications do occur, then the versions that were relevant at the time of processing should be stored for use in data retrieval.
- speed of collection: Speed of collection should be considered a critical factor. In order to analyze the reasons for any abnormal conditions, it is important that the system be capable of recording the events and actions in the precise order in which they occurred.

Get more FREE standards from Standard Sharing Group and our chats

8.5.2.4 Batch and material tracing

The collection of batch histories can support batch and material tracing if it has a complete overview of the batches, including the equipment utilized and the identification of raw materials.

Batch history provides backwards tracing if a certain end product batch history can be traced back to all involved processes, equipment, and ingredients (and to the involved processes, equipment, and ingredients of these ingredients). Forward tracing is available if the consequences of a certain event or the usage of a certain raw material can be traced to all end products affected.

8.5.2.5 Logging from process cell management

Process Cell Management logging should include information associated with initiating and routing the batch, and the equipment-independent information associated with the batch. This includes

- master recipe: the master recipe from which the control recipe was derived — either in copy or by reference. In case of reference, the master recipe should be maintained unchanged as long as the reference may be called.
- Process Cell Management events and control recipe information: information on any changes to and the execution of the control recipe. This includes information such as equipment allocation, start times for batches and unit procedures, and changes in batch states such as pausing or restarting.

- operator comments: narrative descriptions or comments based on the operators' observations of the batch processing. This information entry should be capable of being recorded with the operator's identification.

8.5.2.6 Logging from unit supervision and process control

This data could be dedicated to a single batch or to several batches, such as data from shared-use resources, utility systems, etc. In the latter case the data should be available to all the required batch histories. This includes:

- continuous data: Continuous data is defined as process data that is collected independent of specific events within the batch, with the purpose of giving an accurate history of that measurement.
- pre-specified batch data: data that is specified to be logged during execution of the control recipe. The specification of this data may come from the recipe or be pre-configured. This would include such things as total feed to a reactor or mixing time.
- predictable events: events that are expected to occur, such as start and stop times of procedural elements.
- unpredictable events: Unpredictable event data is defined as a single point entry based on a unpredictable process or physical condition within the batch. This includes such items as process alarms, equipment failures or other upset conditions. In the case of process alarms, the historical data may include the following:
    - Time of activation
    - Time of acknowledgment
    - Time of disappearance of the alarm condition
    - Alarm limit
    - Maximum deviation while the alarm is active
    - Trending information while the alarm is active
- operator interventions: any operator intervention that may affect the processing of the batch. The operator intervention typically is logged with the following information:
    - Intervention type
    - Operator ID
    - Time/date of action

8.5.2.7 Late entries

Late entry data is data entered after execution of the part of the control recipe procedure to which it is related, or after production of the batch. This is typically data that is related to off-line measurements or analyses. Manage batch history includes the logging of such entries, including establishing the link to the associated batch events (like sampling). The following data may be associated with late entries:

- Measured value(s)
- Operator ID
- Lab technician ID
- Time of entry
- Time of sample

8.5.3 Manipulating historical data

The following functions are typical:

- Data manipulation: altering (if legal) or supplementing archived batch data.
- Calculations: perform calculations on batch data creating new batch data related to one batch.
- Data reduction: data reduction on batch history information that is especially relevant with trend information. Loss of data in connection with data reduction should be well defined and related to the dynamics of the data, as well as the requirements of information based on this data.
- Batch tracking information: establishing or maintaining links between batch histories corresponding to the physical movements of the batches, ranging from the use of one batch as raw material to another, to the splitting or combining of batch histories due to splitting or combining of batches.

8.5.4 Producing batch reports

8.5.4.1 Introduction

In this clause any export of data — electronically or on paper — is designated a report.

A batch report is, in general, made on a specific request. Such a request should be possible without knowledge of equipment and time of production. This is the case when

- the batch ID is used as entry key to access the data, not a piece of equipment;
- timing is relative to identified batch events (start of batch, start of operation, etc.);
- entries are identified in generic, batch-related terms and not in equipment-specific tags.

Get more FREE standards from Standard Sharing Group and our chats

8.5.4.2 Recipients of batch reports

Batch history data may be retrieved on request for a number of reasons:

- Production management: production overview summaries, consumption of raw materials and other resources, lot and batch tracking information
- Recipe management: recipe optimization information, comparison between recipe data and actual values, analysis of correlation across several batches, and comparison of trend information
- Process Cell Management: history of current batches and comparisons with old batches for operator display and process control optimization
- External systems:
    - quality control: statistical process control, compliance with product specifications, GMP (Good Manufacturing Practice) documentation
    - maintenance: alarms, equipment usage documentation
    - financial: raw material consumption, yields, produced quantities, etc.
    - customer support: product documentation
- Internally within manage batch history: Process Cell Management may include functions to perform the queries mentioned above and the ability to export or print them on request, at regular intervals, or after each batch.

8.5.4.3 Elements of batch reports

Some of the possible elements of a batch report include

- report header: This header contains information on the report type, batch or batches displayed in the report, descriptive text, etc.
- single elements: These data elements are displayed somewhere on the paper/screen. This may include calculated values.
- event lists: These are chronological lists of event-type entries with associated data. For example, this might include a list of alarms or a list of operator interventions.
- merging of entries in event lists: Entries with different tags and of different types may be merged into the same list.
- selection of entries into lists: Entries may be selected according to different criteria before entering lists. For example, the entries may include only high priority alarms.
- trends: These displays show one or more values on the same time axis.
    - single batch trend: These are trends that display data from one batch or a portion of a batch . They may display several values with individual time axis. The display may be in relative or absolute time.
    - multi-batch trend: These are trends that compare values from several batches in one trend display. They should be with relative time-axis. Some variables may be normalized to a standard amount.
    - event-marking in trends: Events may be introduced in the trend display by "ticks" on trends or other indications. The tick should refer to a specific event-type entry.
- time-series: These are displays of a time-series of one or more entries in a table-like fashion. The time-deadband, which is how close in time entries with different tags have to be in order to be displayed on the same line, should be specified in time series displays.
- interpolation: Rules for interpolation of data have to be established if data with different entry-times have to be displayed on one line or if the data are used in calculations.

8.6 Process cell management

8.6.1 Introduction

Process Cell Management is the collection of functions that manages all batches and resources within a process cell. Within this control activity, control recipes are created from master recipes, each batch is defined as an entity, individual batches are initiated and supervised, resources within the process cell are managed to resolve conflicts for their use and process cell and batch data are collected. Process Cell Management interfaces with Unit Supervision, Recipe Management, Production Planning and Scheduling, and Production Information Management (see Figure 26).

At the process cell level, there are often multiple batches and multiple units, and each unit may be carrying out a unit procedure for a different batch. The progression of the procedure for each batch and the utilization of the individual pieces of equipment should be coordinated based on information derived from the control recipe, scheduling information, operator input, and status of equipment and other common resources.

The domain of Process Cell Management is the process cell. The successful execution of a control recipe makes a batch, and Process Cell Management is finished with the batch when the control recipe procedure is complete. The batch that has been produced does not have to be a final product. It may take several control recipes running in the same process cell or in different process cells and/or sites to make the finished product(s). When a batch leaves the process cell, it is no longer the responsibility of Process Cell Management associated with that process cell in terms of identification, batch tracking, etc.

Process Cell Management can be discussed in terms of the following three functions (see Figure 26):

- Manage batches
- Manage process cell resources
- Collect batch and process cell information

Through the manage batches and manage process cell resources functions, process cell management delivers each control recipe component at the unit procedure level to unit supervision to carry out its intent in the unit requested by process cell management. In describing these activities, the term unit recipe is used in place of “control recipe component at the unit procedure level.” Unit recipe is defined as that part of a control recipe which uniquely defines the contiguous production requirements for a unit. It includes the unit procedural element and its related formula, header, equipment requirements, and other information.

NOTE A unit recipe is a component of a control recipe, not an additional type of recipe.

8.6.2 Manage batches

This is the function in which a control recipe is created from a copy of a master recipe, a batch is initiated based on the scheduling information and operator input, and the execution of the batch is supervised.

The following capability is typically included in this function:

- Creating a control recipe from the master recipe, scheduling information, and input received from the operator. This may happen with widely varying lead times, such as at the instant needed in some situations and well in advance of the scheduled execution time in others. The control recipe may be created initially in its entirety, or it may be created incrementally as the information is needed.
- Assigning a unique batch identification (batch ID) to each batch and to the associated control recipe. A batch may be identified or named in many different ways, but at least one identification, referred to here as the batch ID, should be verified to be totally unique within the process cell at any given time. The batch ID may be provided by the operator, in the scheduling information, or from within Process Cell Management, but uniqueness is typically verified before it is associated with a batch.

Get more FREE standards from Standard Sharing Group and our chats

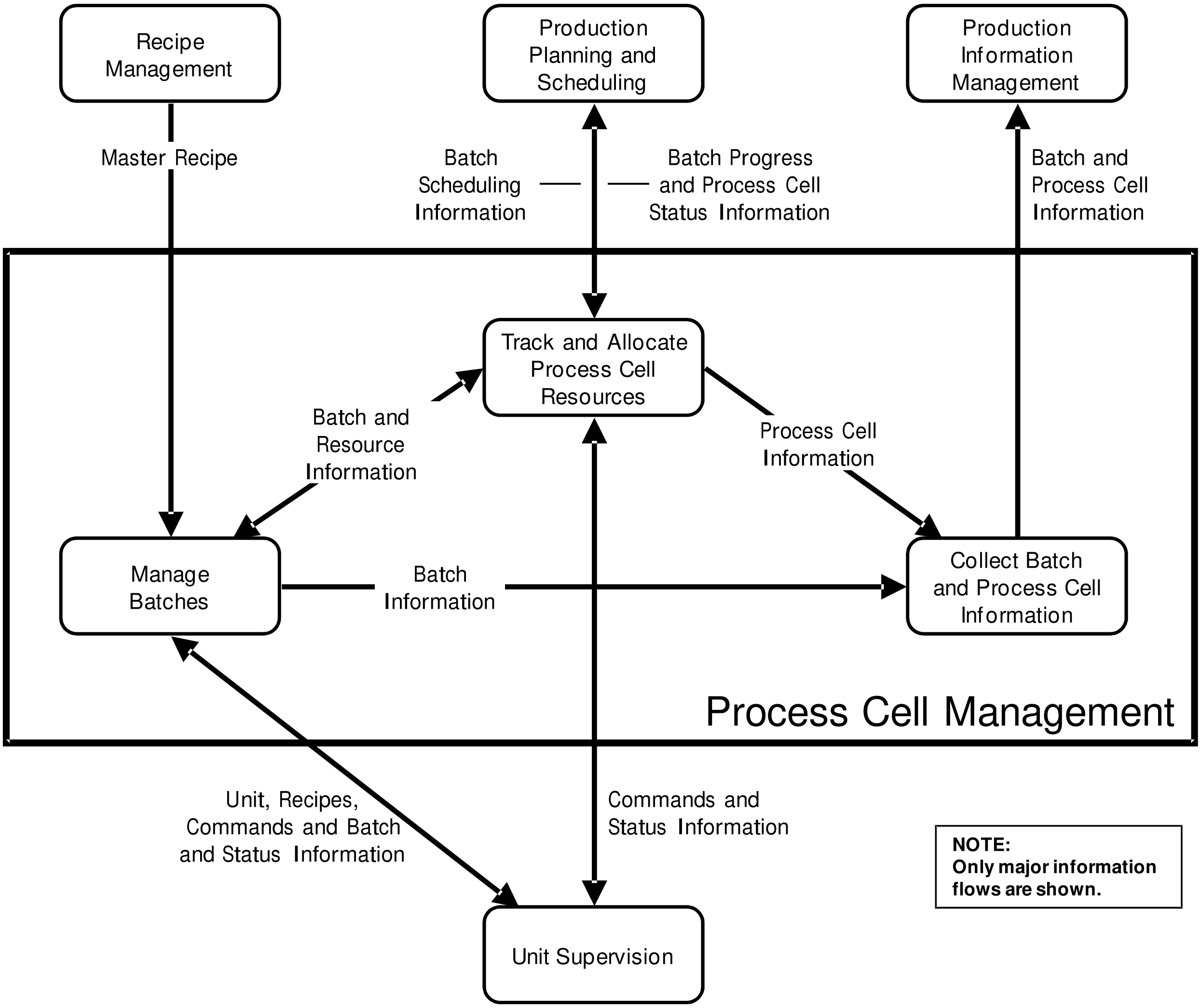

Recipe Management
Production Planning and Scheduling
Production Information Management
Master Recipe
Batch Scheduling Information
Batch Progress and Process Cell Status Information
Batch and Process Cell Information
Track and Allocate Process Cell Resources
Batch and Resource Information
Process Cell Information
Manage Batches
Batch Information
Collect Batch and Process Cell Information
Process Cell Management
Unit, Recipes, Commands and Batch and Status Information
Commands and Status Information
NOTE: Only major information flows are shown.
Unit Supervision

Figure 26 — Process cell management

- Verifying the control recipe as it is created. Verifying consists of ensuring that the control recipe is complete and is executable on the selected set of units. This includes verifying that all procedural elements are available, that formula information is valid, and that necessary resources can be expected to be available when needed.
- Sizing the control recipe to meet the batch quantity needed based on the sizing rules in the master recipe and the quantity specified in the batch scheduling information. The recipe may include the range over which it may be scaled.
- Maintaining all the current control recipes within Process Cell Management until the batches are completed.
- Assigning the start conditions as specified in the scheduling information and/or provided by the operator. Some batch start conditions that may be used, either individually or in some combination, include the following:
  - Start batch as soon as a unit becomes available
  - Start based on operator direction
  - Start when specific units are available

  - Start based on the scheduled priority of the batch
  - Start based on a scheduled time

- Modifying any part of a control recipe that has not been executed. This may include the ability to modify the procedure, such as adding and deleting unit procedures, operations, and/or phases, or looping back to repeat unit procedures, operations, and/or phases that have previously been executed.
- Requesting and releasing units and other equipment, changing their status to indicate use, and updating the manage process cell resources function on the status of the batch.
- Monitoring and controlling the executing control recipe(s) including the current status of the batch, such as what unit procedures have been executed, and what unit procedure is next.
- Processing requests for state and mode changes to procedures, unit procedures, operations, and phases.
- Allowing a control recipe to span multiple units in the same process cell, including distributing unit recipes to Unit Supervision in a timely manner.
- Allowing a batch to be suspended, removed from the processing equipment (packaged for temporary storage), and therefore out of the control of Process Cell Management, and later recalled to complete the batch processing.
- Maintaining batch status information. The control recipe, including all modifications, should be logged as part of the batch history as it is executed or at least when the batch leaves the process cell.
- Making information on batches available to the collect batch and process cell information function.

8.6.3 Track and allocate process cell resources

Get more FREE standards from Standard Sharing Group and our chats

This is an execution related function in which process cell resources are made available for execution by allocating and reserving units and other equipment, by arbitrating multiple requests for the same equipment, and by providing a mechanism for maintaining status of both allocated and unallocated equipment. Process cell resources also include the materials within the process cell. Process cell resource tracking and allocation should maintain information including which materials are in the process cell, which are pre-allocated by the batch schedule, the materials that are available for allocation during recipe execution, the units that contain them, and their disposition.

An assignment of resources at the process cell or unit level (resource allocation) needs to be provided in order for Process Cell Management to be able to assign the equipment or equipment options according to active recipes being executed and the batch schedule. Although limited in function to execution time allocations, limited equipment reassignment and generation of a new resource allocation at the process cell or unit level may also be needed by an operator. This new resource allocation may be necessary because of such variables as an execution-time malfunction in equipment or availability of raw materials that could not be anticipated by production planning and scheduling. Production Planning and Scheduling may require notification of this new resource allocation to allow for assessment of impact.

The following capabilities are typically included in this function:

- Obtaining scheduling information from Production Planning and Scheduling and providing this information to the manage batches function
- Allocating or reserving equipment as requested by the manage batches function. Within a process cell, batches may move from unit to unit. In each unit a portion of the control recipe, corresponding to the unit procedure, is executed. The control of what equipment to allocate

to the different batches, and when transfers can take place may require control at the process cell level.

EXAMPLE Some examples of how this allocation may be done are

- according to a batch schedule designating each individual unit allocation; or
- according to a strategy defined at the process cell level combining the equipment requirements of the control recipe and the availability and capabilities of equipment at the time of execution.

- Arbitrating, as required, multiple requests for reservation or allocation of the same equipment during control recipe execution. The rules for arbitration may be simple or complex, depending on the application.

  EXAMPLE Examples of arbitration rule sets include the following:

  - Order of request (FIFO)
  - Timed requests (such as by reserving the equipment)
  - Priority of batch
  - Maximizing equipment utilization (such as by minimizing cleaning requirements, minimizing energy consumption, or maximizing throughput)
  - Operator judgment

- Keeping track of both allocated and unallocated equipment within the process cell
- Receiving status information sent by Unit Supervision and/or status information sent by Process Control related to unallocated equipment within the process cell
- Updating information on all process cell resources to the collect batch and process cell information function
- Updating Production Planning and Scheduling with batch progress information, such as
  - batch ID;
  - batch state change events;
  - actual quantities of raw materials, products, and utilities;
  - equipment assignments; and
  - projected and actual allocation and de-allocation times of process cell resources.

8.6.4 Collect batch and process cell information

This is the function in which information is collected about Process Cell Management events, both batch and equipment oriented, from the manage batches and manage process cell resources functions. This information is made available to Production Information Management.

EXAMPLE Examples of the types of information collected include the following:

- Mode and state changes
- Incremental copies of control recipes as each portion is completed
- Time that commands were sent to Unit Supervision and Process Control
- Time that unit recipes were sent to Unit Supervision
- Delays encountered due to lack of equipment availability
- Time of allocation, reservation and release of each process cell resource

- Requests and result of requests for equipment allocation or reservation which required arbitration
- Status changes in unallocated equipment
- Operator intervention

8.7 Unit supervision

8.7.1 Introduction

Unit Supervision is the control activity that ties the recipe and procedural control to equipment via Process Control (see Figure 27). This control activity interfaces with Process Cell Management, Process Control, and Production Information Management. There are three main functions within this control activity that are discussed in this clause. They include acquiring and executing procedural elements, managing unit resources, and collecting batch and unit information.

8.7.2 Acquire and execute procedural elements

Process Cell Management supplies the unit recipe that will be executed within the unit and also supplies other batch information required to manufacture the batch.

Get more FREE standards from Standard Sharing Group and our chats

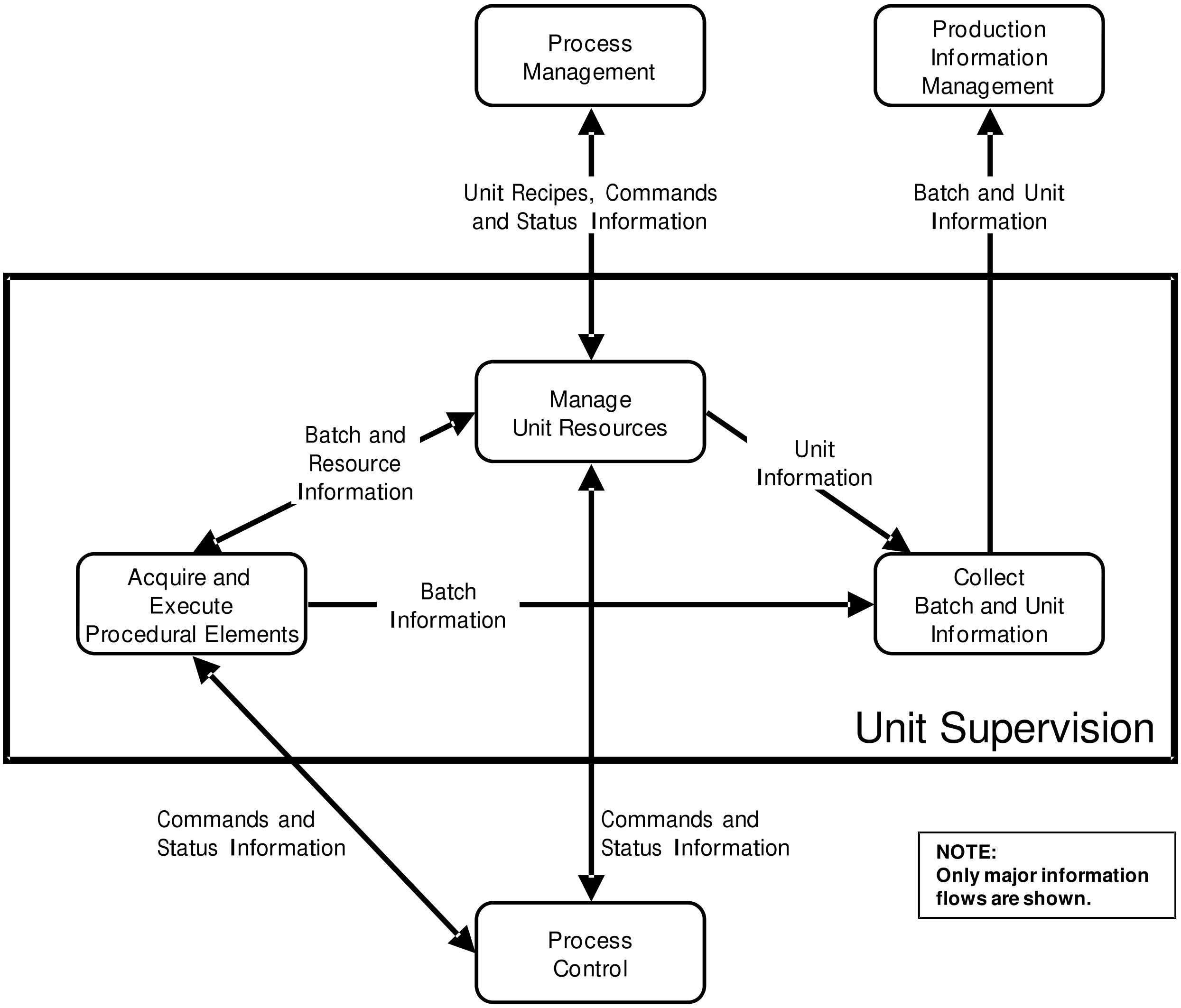

Process Management
Production Information Management
Unit Recipes, Commands and Status Information
Batch and Unit Information
Manage Unit Resources
Batch and Resource Information
Unit Information
Acquire and Execute Procedural Elements
Batch Information
Collect Batch and Unit Information
Unit Supervision
Commands and Status Information
Commands and Status Information
NOTE: Only major information flows are shown.
Process Control

Figure 27 — Unit supervision

Unit Supervision has to be able to determine from the unit recipe the procedural logic to be run, the appropriate parameters, the equipment entities to be utilized, and other pertinent information, such as the name of the product, equipment restrictions, and the batch identification.

Acquire and execute procedural elements includes the execution of unit procedures. If the unit procedure is part of equipment control in the unit, this function associates the recipe unit procedure, including the parameters, with the equipment unit procedure. Initiation and parameterization of operations (as either equipment or recipe operations) is part of the execution of a unit procedure.

Acquire and execute procedural elements includes the execution of operations. If the operation is part of equipment control in the unit, this function associates the recipe operation, including the parameters, with the equipment operation. The initiation and parameterization of phases (as either equipment or recipe phases) is part of the execution of an operation.

Acquire and execute procedural elements includes the initiation and/or execution of equipment procedural control at the phase level. There are two possible conditions:

- If the equipment phase is part of equipment control in the unit, this function shall associate the recipe phase, including the parameters, with the equipment phase then execute the equipment phase.
- If the equipment procedural control to be initiated is part of equipment control in an equipment module, this function shall initiate and parameterize the required sequences, but its execution will be handled by the equipment module as part of the process control activity (see Clause 8.8).

When executing equipment procedural control in a unit, this function may affect the process through commands and parameters it sends to equipment modules and control modules, either directly or indirectly, but does not have the capability of directly interfacing with physical equipment.

The following capabilities are typically included in this function:

- Determining which procedural elements are to be executed
- Verifying that the procedural elements exist
- Executing unit procedures, operations, and phases, except when part of equipment control in an equipment module
- Associating recipe procedural elements with equipment procedural elements
- Initiating and parameterizing equipment procedural elements
- Manual intervention into the execution of procedural elements in the unit.

8.7.3 Manage unit resources

This function includes managing resources that are part of the unit, initiating requests for resources that are not part of the unit, managing resources that have been acquired and not yet released, initiating requests for services from other units, and providing services to other units.

Get more FREE standards from Standard Sharing Group and our chats

During the execution of a recipe, it may be necessary to acquire the services of shared-use and/or exclusive-use common resources that will subsequently be released. Units can request services from or provide services to other units. The phases or operations in the units can communicate to perform a coordinated function.

Unit-to-unit coordination may be used to enable functions such as material transfers between units.

The following capabilities are typically included in this function:

- Issuing requests to, reacting on feedback from, and interfacing with arbitration functions related to the equipment in question
- Ensuring appropriate propagation of unit and procedural element modes and states
- Enabling collection of production information relevant to the batch from external equipment

8.7.4 Collect batch and unit information

The Collect Batch and Unit Information function makes information available to Production Information Management about Unit Supervision events, both batch and equipment oriented.

Data collection may be conditional. That is, certain data might not always be collected or might be sampled at a different time interval, depending upon information received from another function, such as from parameters passed to the equipment procedural element.

EXAMPLE Examples of the types of information collected include the following:

- Mode and state changes
- Timing of commands sent to Process Control
- Timing of the execution of the unit recipe procedure events
- Timing and sequence of allocation, reservation, and release of equipment entities acquired by the unit
- Status changes in unit equipment
- Values derived during execution of the unit recipe

8.8 Process control

8.8.1 Introduction

This control activity encompasses procedural and basic control, including sequential, regulatory, and discrete control, in addition to gathering and displaying data. This control activity will be distributed among equipment entities at the equipment module and control module levels. It interfaces with Production Information Management, Unit Supervision, and Personnel and Environmental Protection.

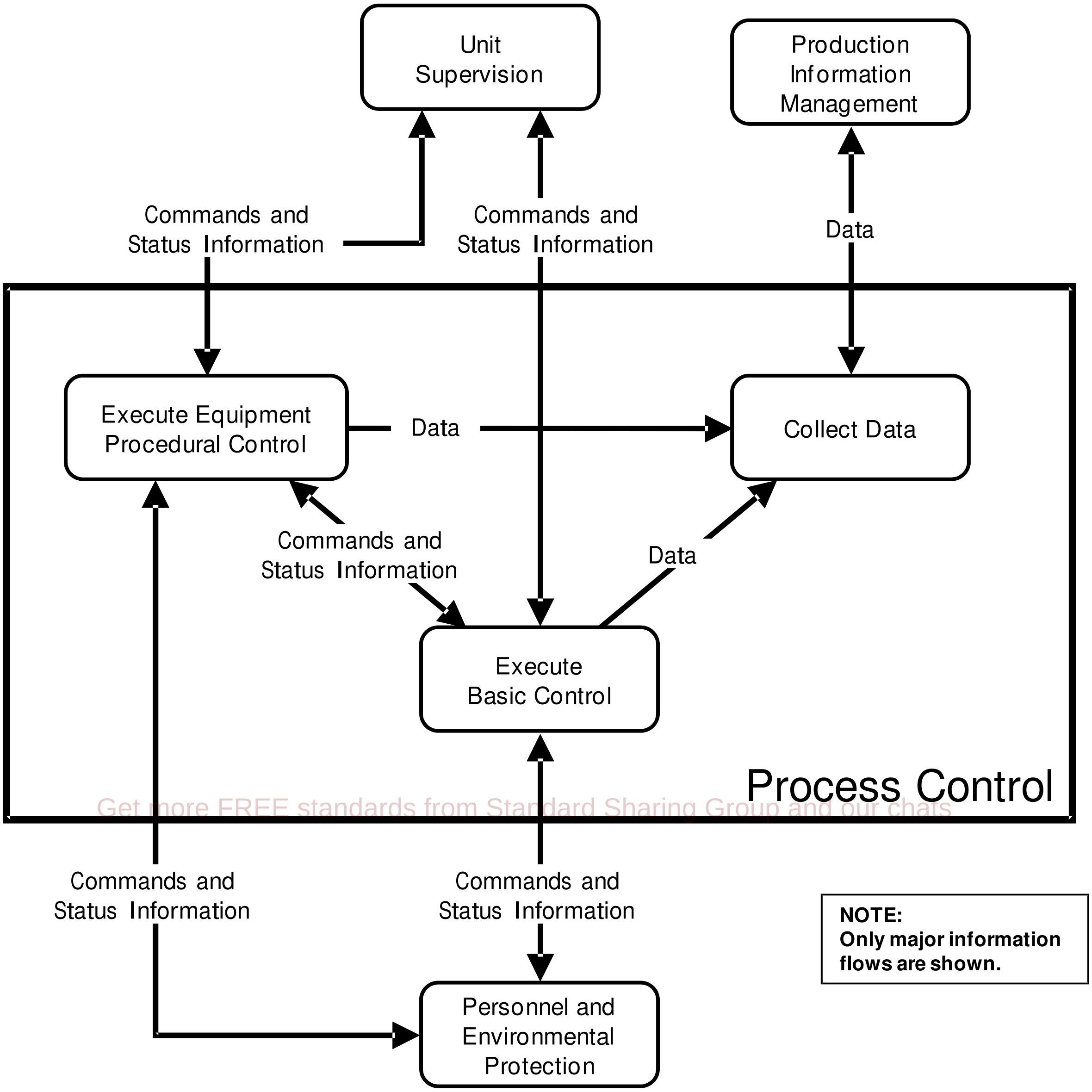
Process Control can be discussed in terms of three functions: execute equipment procedural element, execute basic control, and collect data (see Figure 28).

8.8.2 Execute procedural control

This function is initialized by and receives necessary parameter values from the *Acquire and execute procedural element* function in Unit Supervision (see Clause 8.7.2). The *Execute procedural control* function interprets the procedure initialization command and associates the necessary parameters with the appropriate level of equipment procedural control contained in equipment entities. The equipment procedural control in this case may be commanded and parameterized before or during execution.

NOTE When the equipment phases are part of equipment control in a unit, the activities defined here for equipment modules will be performed by the *acquire and execute procedural elements* function of Unit Supervision. In this case Unit Supervision provides commands directly to basic control.

This function does not act directly on physical equipment. It influences the process only through the basic control in equipment modules and control modules.



Get more FREE standards from Standard Sharing Group and our chats

Figure 28 — Process control

This function also includes supervision of the modes and states of equipment procedural control in equipment modules. This includes

- the propagation of modes and states from or to the unit or equipment module initiating execution; and
- manual intervention into the execution of the equipment module.

8.8.3 Execute basic control

Executing basic control is a function that causes changes in equipment and process states by sending commands to actuators and other control modules. Commands to basic control may come from the execution of an equipment procedural control or from another function, such as a manual command from an operator. Basic control uses input from sensors and other functions in order to execute its function. The execution of this function may also result in process, equipment, and other status information being provided to high level functions. Some other basic functions that may be included are execution of state transition sequences when required, exception handling, calculations, and treatment of operator-entered information, etc.

However, the execute basic control function shall not contain procedural or equipment procedural element control and is always configured as part of the equipment entity. This function also includes the association of the necessary parameters with the appropriate basic control function. The only equipment entities directly capable of performing this function are equipment modules and control modules. In general, units and process cells do not directly perform the function of execute basic control. These entities may normally accomplish basic control by making use of contained or referenced equipment modules and control modules to perform this control function.

This function also includes the supervision of equipment entity modes and states. This includes

- the propagation of modes and states from/to any equipment entities and/or procedural elements; and
- manual intervention.

Where the equipment entity is a common resource, this function may also be involved in the arbitration of conflicting requests and commands.

8.8.4 Collect data

In the Collect Data function, data from sensors, derived values, and events that occur within the domain of Process Control are collected and stored in batch history. Data collection may be conditional. That is, certain data might not always be collected or might be sampled at a different time interval, depending upon information received from another function, such as from parameters passed to the equipment phase.

8.9 Personnel and environmental protection

The Personnel and Environmental Protection control activity provides safety for people and the environment. It is shown in the Control Activity Model in Figure 24 (see Clause 8.2) below Process Control because no other control activity should intervene between Personnel and Environmental Protection, and the field hardware it is designed to operate with. Personnel and Environmental Protection is, by definition, separate from higher level control activities. It may map to more than one level of equipment entity if that level of organization or sophistication is required to provide adequate safety protection.

Personnel and environmental protection is included in the control activity model to emphasize the importance of these types of protection systems and to indicate the point in the model appropriate for insertion of a separate protection system of this type. A complete discussion of personnel and environmental protection, the classification of these types of systems, and the segregation of levels of interlocks within these systems is a topic of its own and beyond the scope of this standard. More information on this topic can be obtained from some of the standards and guidelines that are under development (see References 1, 2, 3, 4, and 5 in Annex E).

9 Completeness, compliance and conformance

9.1 Completeness

The number of models in this Part that a specification or implementation complies with, or that an implementation tool (e.g. software or system) conforms to, shall determine its degree of completeness. Models in this Part (and the primary definitions of each) are:

- Process Model (see Figure 2 and Clause 4.3)
- Physical Model (see Figure 3 and Clause 4.4.2)
- Equipment Entity Model (see Figure 4 and Clause 4.4.3)
- Procedural Control Model (see Figure 9 and Clause 5.3.2)
- Recipe Types Model (see Figure 11 and Clause 6.2)
- General Recipe Procedure Model (see Figure 13 and Clause 6.5.2)
- Master Recipe Procedure Model (see Figure 14 and Clause 6.5.4)
- Control Activity Model (see Figure 24 and Clause 8.2)

9.2 Compliance

Compliance, as it is to be understood here, is measured by documented assessment of an implementation or the specification of an implementation to determine whether it is in alignment with a specified standard.

Any assessment of the degree of compliance shall be qualified by the following:

Get more FREE standards from Standard Sharing Group and our chats

- the use of the terminology and model defined in this Part;
- a statement of the degree to which the terminology used complies partially or totally to the definitions in this Part;
- a statement of the degree to which the models used complies partially or totally to the models in this Part, its text describing each of their required features and capabilities, and its text describing each optional feature or capability that relates to the specification or implementation;
- in the event of partial compliance, areas of non-compliance shall be explicitly identified;
- a complete statement of the criteria and protocols used in the assessment.

NOTE This part of the standard does not enumerate compliance points sufficient to form a conformity assessment scheme. However, an informative discussion in Annex C.1 is included to provide some example guidance in assessing compliance.

9.3 Conformance

Conformance, as it is to be understood here, is measured by documented assessment of an implementation tool (e.g. software or system) or the specification of an implementation tool to determine whether it meets some specified standard.

Any assessment of the degree of conformance shall be qualified by the following:

- the use of the terminology and models defined in this Part;
- a statement of the degree to which the terminology used conforms partially or totally to the definitions in this Part;

- a statement of the degree to which the modelling capabilities provided conform partially or totally to the models in this Part and its text describing each of their required or optional features and capabilities;
- in the event of partial conformance, areas of non-conformance shall be explicitly identified
- A complete statement of the criteria and protocols used in the assessment.

NOTE This part of the standard does not enumerate or group the conformance points sufficient to form a conformity assessment scheme. However, an informative discussion in Annex C.1 is included to provide some example guidance in assessing conformance.

This page intentionally left blank.

Get more FREE standards from Standard Sharing Group and our chats

Annex A
(informative)

Model philosophy

A number of drawing formats are used in this standard. Each of these drawing formats is discussed below.

The model formats discussed in this clause provide a non-rigorous method of portraying information and relationships. This standard does not intend to recommend or imply an analysis methodology or to have the figures supersede the information described in the text.

— Physical drawings use the ISA symbol standards, where applicable. Figure 6 is an example.

— The process model, physical model, equipment entity model, procedural control model, general recipe procedure model, and master recipe procedure model are illustrated using instance diagrams, of which Figure 9 is an example.

— Model element instances are shown as rectangles in each instance diagram. The exception to this rule occurs at the lowest implemented level of each recipe, where a different symbol is used to highlight that the processing or procedural information for that level is defined outside of the recipe. Figure 16 is an example.

— Labelled associations in each instance diagram illustrate the relationships between model elements. For example, Figure 2 illustrates that in the process model each Process Stage “specifies the order to process one or more” Process Operations.

— Nested drawings are used in Figure 1 and Figure 12 to show containment or encapsulation type relationships.

— Each rounded rectangle in the example state diagram for procedural elements (see Figure 23) represents a single procedural state. Lines between states or collections of states identify state changes that may be initiated automatically or by the indicated command.

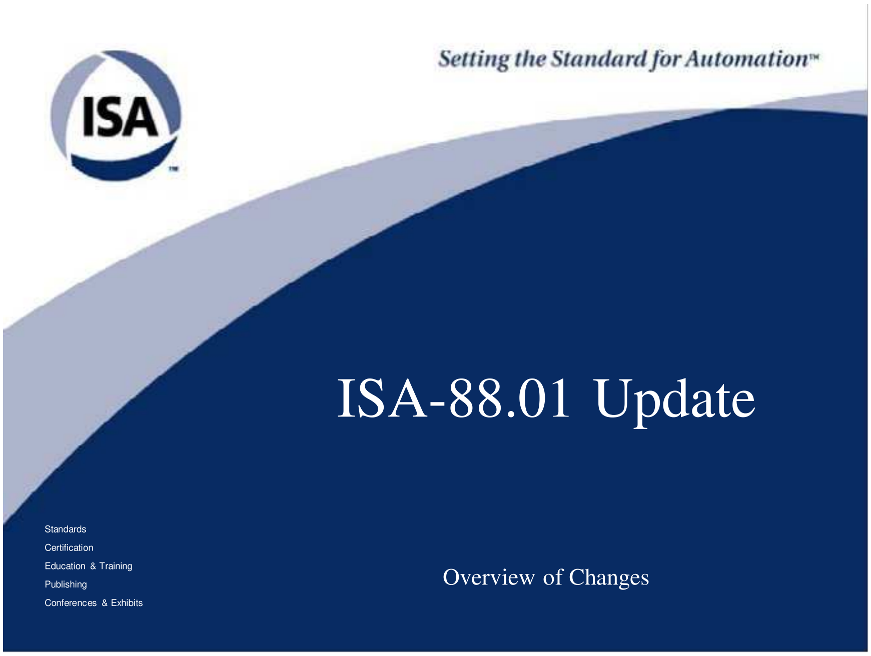
— Activities or functions are shown as rounded rectangles in the relationship diagrams used to illustrate the Control Activity Model. Figure 25 through Figure 28 only show the explosion of one control activity per diagram. Lines between activities and between functions show information exchange. The Process Control activity as shown in Figure 28 is an example.

This page intentionally left blank.

Get more FREE standards from Standard Sharing Group and our chats

Annex B
(informative)

Overview of Part 1 Update

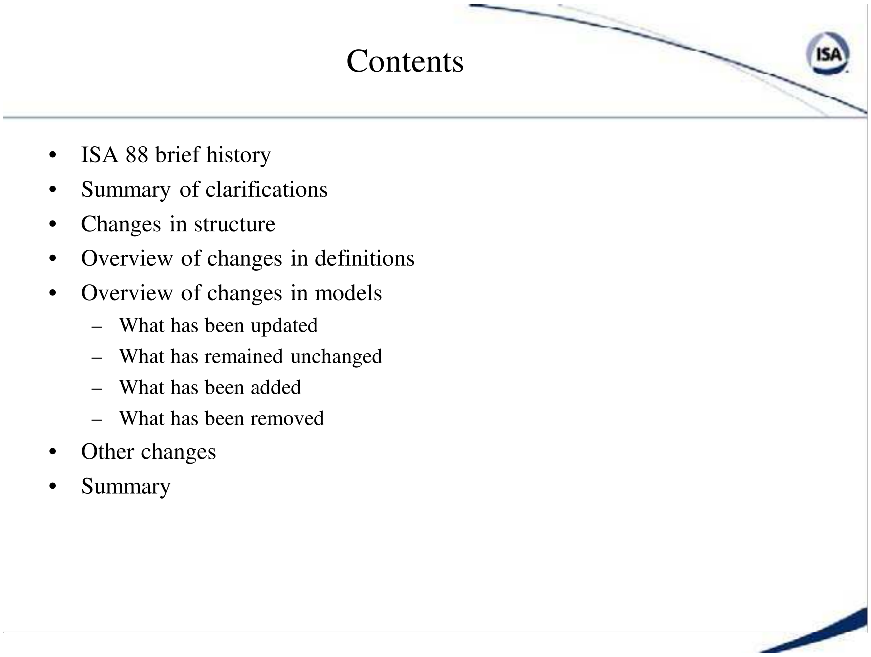


Setting the Standard for Automation™

ISA-88.01 Update

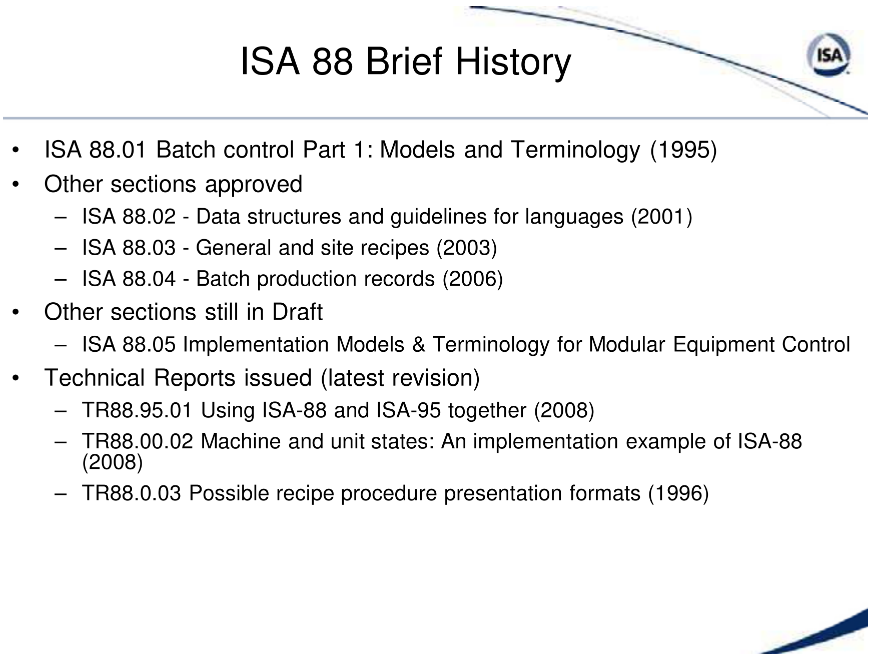
Standards
Certification
Education & Training
Publishing
Conferences & Exhibits

Overview of Changes



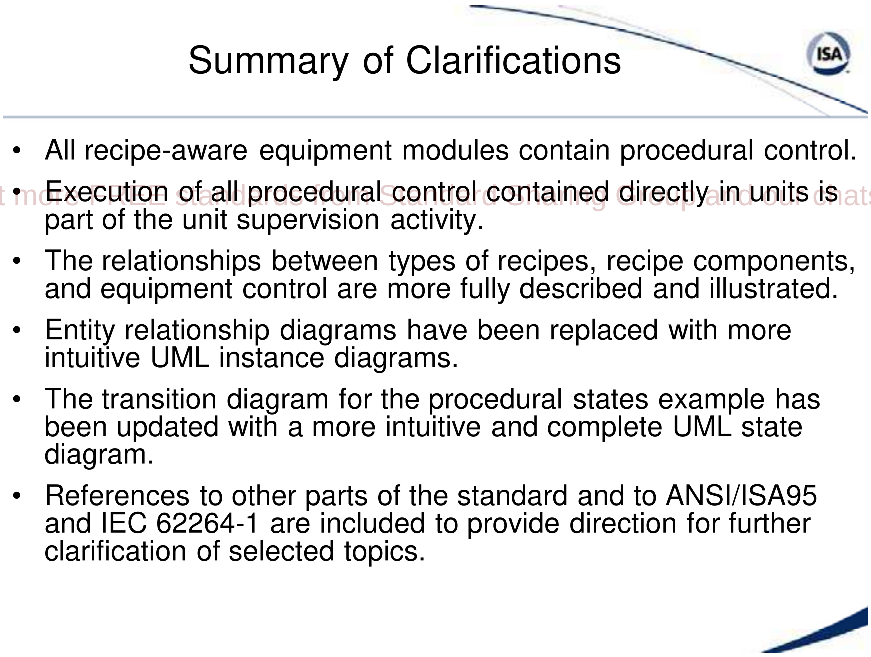
Contents

- ISA 88 brief history
- Summary of clarifications
- Changes in structure
- Overview of changes in definitions
- Overview of changes in models
  - What has been updated
  - What has remained unchanged
  - What has been added
  - What has been removed
- Other changes
- Summary



ISA 88 Brief History

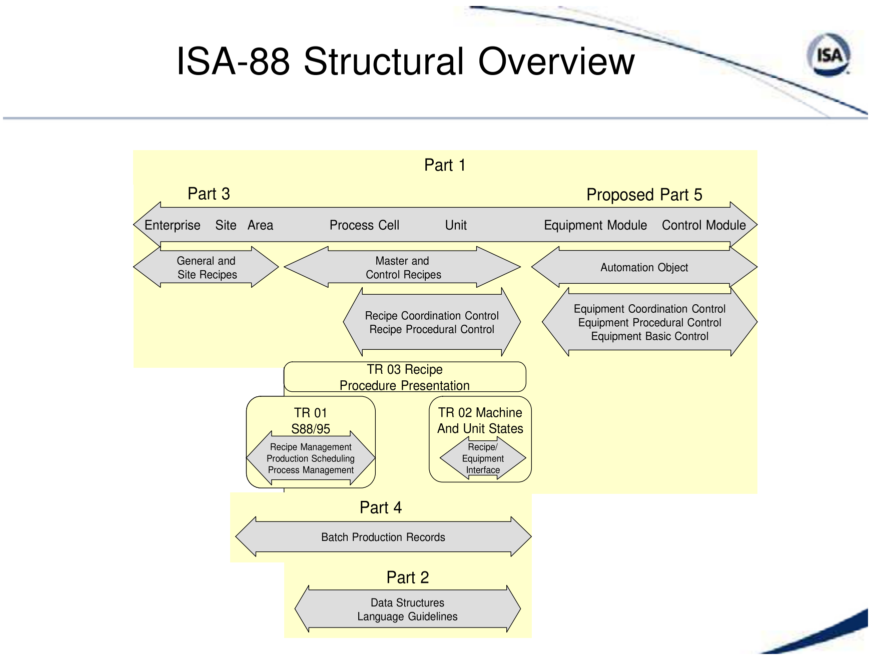
- ISA 88.01 Batch control Part 1: Models and Terminology (1995)
- Other sections approved
  - ISA 88.02 - Data structures and guidelines for languages (2001)
  - ISA 88.03 - General and site recipes (2003)
  - ISA 88.04 - Batch production records (2006)
- Other sections still in Draft
  - ISA 88.05 Implementation Models & Terminology for Modular Equipment Control
- Technical Reports issued (latest revision)
  - TR88.95.01 Using ISA-88 and ISA-95 together (2008)
  - TR88.00.02 Machine and unit states: An implementation example of ISA-88 (2008)
  - TR88.0.03 Possible recipe procedure presentation formats (1996)



Summary of Clarifications

- All recipe-aware equipment modules contain procedural control.
- Execution of all procedural control contained directly in units is part of the unit supervision activity.
- The relationships between types of recipes, recipe components, and equipment control are more fully described and illustrated.
- Entity relationship diagrams have been replaced with more intuitive UML instance diagrams.
- The transition diagram for the procedural states example has been updated with a more intuitive and complete UML state diagram.
- References to other parts of the standard and to ANSI/ISA95 and IEC 62264-1 are included to provide direction for further clarification of selected topics.

Get more FREE standards from Standard Sharing Group and our chats



ISA-88 Structural Overview

Part 1

Part 3

Proposed Part 5

Enterprise Site Area Process Cell Unit Equipment Module Control Module

General and Site Recipes

Master and Control Recipes

Automation Object

Recipe Coordination Control
Recipe Procedural Control

Equipment Coordination Control
Equipment Procedural Control
Equipment Basic Control

TR 03 Recipe
Procedure Presentation

TR 01
S88/95

Recipe Management
Production Scheduling
Process Management

TR 02 Machine
And Unit States

Recipe/
Equipment
Interface

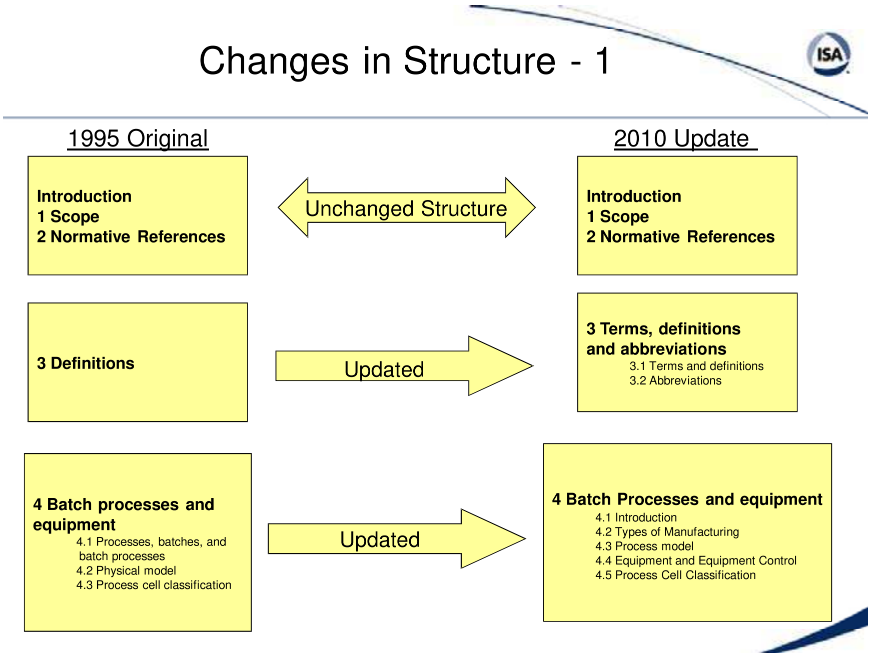
Part 4

Batch Production Records

Part 2

Data Structures
Language Guidelines

With the additional parts and technical reports written, approved, and adapted, the Part 1 revision now references the other parts and technical reports for details and clarification.



Changes in Structure - 1

1995 Original

2010 Update

Introduction
1 Scope
2 Normative References

Unchanged Structure

Introduction
1 Scope
2 Normative References

3 Definitions

Updated

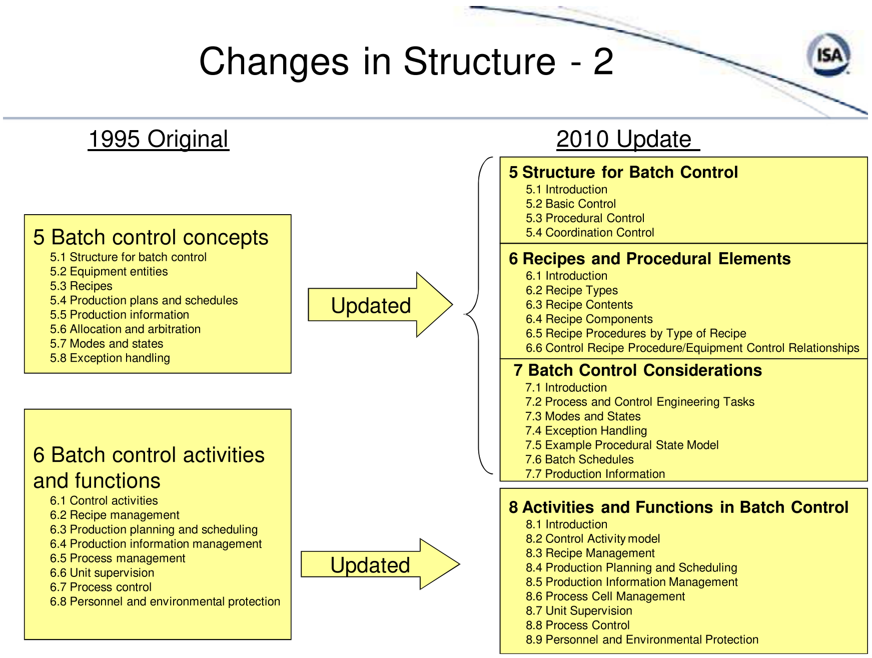
3 Terms, definitions and abbreviations
3.1 Terms and definitions
3.2 Abbreviations

4 Batch processes and equipment
4.1 Processes, batches, and batch processes
4.2 Physical model
4.3 Process cell classification

Updated

4 Batch Processes and equipment
4.1 Introduction
4.2 Types of Manufacturing
4.3 Process model
4.4 Equipment and Equipment Control
4.5 Process Cell Classification

The ISA88 Part 1 revision begins with an equivalent structure to the original. Clause 4.1 has been separated into 2 subclauses covering Types of Manufacturing and the Process model. Recognizing that segmentation of the process cell in the physical model is determined by the logical scope of control for equivalent equipment entities (the real objects of interest to ISA-88), Clause 4.2 has been interwoven with parts of 5.2 to form a consolidated clause titled Equipment and Equipment Control.



Changes in Structure - 2

1995 Original

5 Batch control concepts
- 5.1 Structure for batch control
- 5.2 Equipment entities
- 5.3 Recipes
- 5.4 Production plans and schedules
- 5.5 Production information
- 5.6 Allocation and arbitration
- 5.7 Modes and states
- 5.8 Exception handling

6 Batch control activities and functions
- 6.1 Control activities
- 6.2 Recipe management
- 6.3 Production planning and scheduling
- 6.4 Production information management
- 6.5 Process management
- 6.6 Unit supervision
- 6.7 Process control
- 6.8 Personnel and environmental protection

2010 Update

5 Structure for Batch Control
- 5.1 Introduction
- 5.2 Basic Control
- 5.3 Procedural Control
- 5.4 Coordination Control

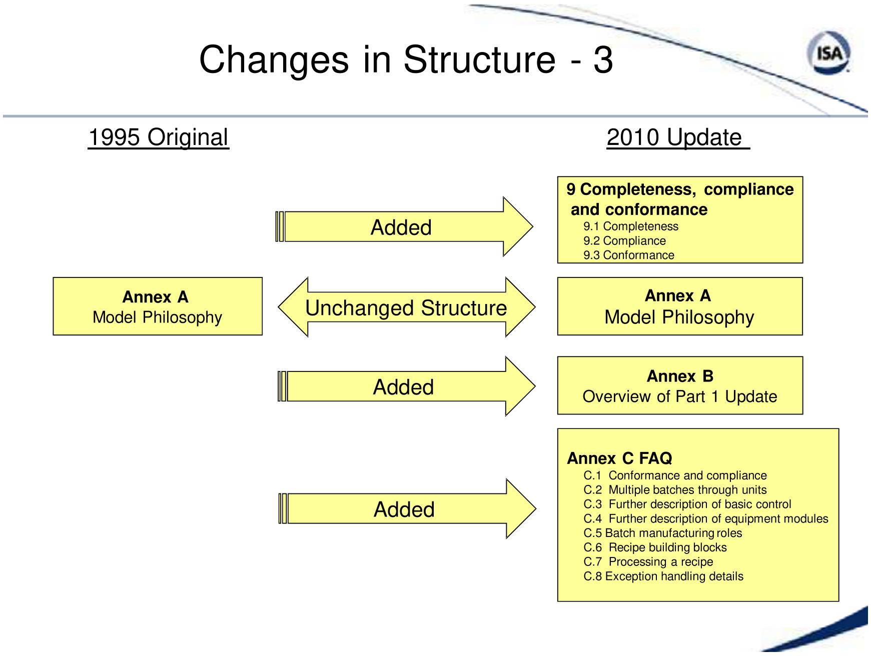
6 Recipes and Procedural Elements
- 6.1 Introduction
- 6.2 Recipe Types
- 6.3 Recipe Contents
- 6.4 Recipe Components
- 6.5 Recipe Procedures by Type of Recipe
- 6.6 Control Recipe Procedure/Equipment Control Relationships

7 Batch Control Considerations
- 7.1 Introduction
- 7.2 Process and Control Engineering Tasks
- 7.3 Modes and States
- 7.4 Exception Handling
- 7.5 Example Procedural State Model
- 7.6 Batch Schedules
- 7.7 Production Information

8 Activities and Functions in Batch Control
- 8.1 Introduction
- 8.2 Control Activity model
- 8.3 Recipe Management
- 8.4 Production Planning and Scheduling
- 8.5 Production Information Management
- 8.6 Process Cell Management
- 8.7 Unit Supervision
- 8.8 Process Control
- 8.9 Personnel and Environmental Protection

The core control concepts described by Clauses 5.1 (with corresponding parts of 5.2) and 5.3 were promoted to Clauses 5 and 6, respectively. The remainder of Clause 5, together with 6.3 and 6.4, form the new Clause 7 that supports the models, but is not part of their core definitions. The remainder of Clause 6 forms the new Clause 8.

Get more FREE standards from Standard Sharing Group and our chats



Changes in Structure - 3

1995 Original

2010 Update

Added

9 Completeness, compliance and conformance
9.1 Completeness
9.2 Compliance
9.3 Conformance

Annex A
Model Philosophy

Unchanged Structure

Annex A
Model Philosophy

Added

Annex B
Overview of Part 1 Update

Added

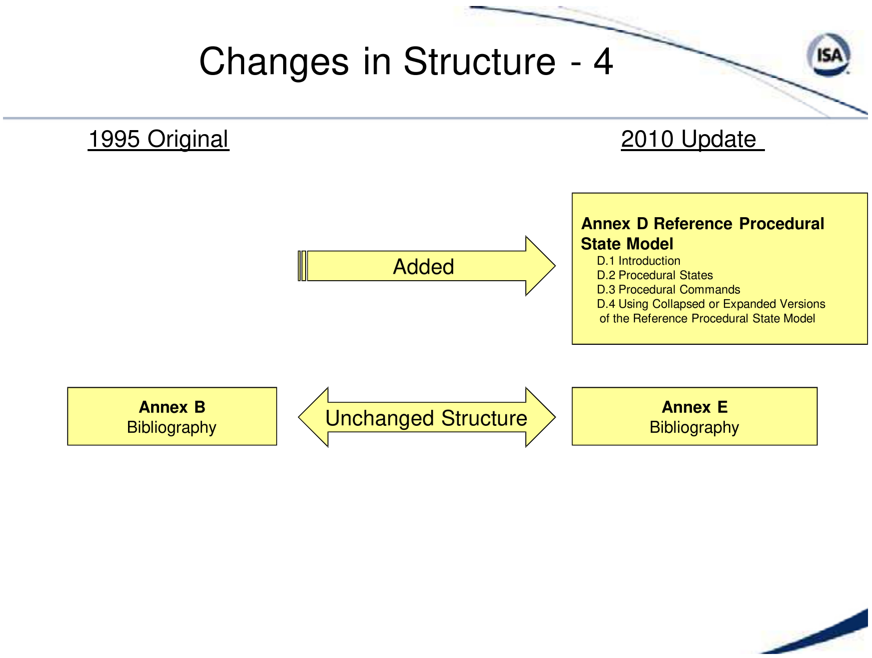
Annex C FAQ
C.1 Conformance and compliance
C.2 Multiple batches through units
C.3 Further description of basic control
C.4 Further description of equipment modules
C.5 Batch manufacturing roles
C.6 Recipe building blocks
C.7 Processing a recipe
C.8 Exception handling details

Clause 9 is entitled Completeness, Compliance, and Conformance and is normative. The intent of this clause is to define compliance and conformance relative to the normative models and terminology in this part of the standard.

The original Annex A has been updated to reflect the new figure styles used throughout the draft. It is now informative rather than normative.

The New Annex B (this overview) is informative.

The New Annex C FAQ is informative. It provides clarification of a number of frequently misunderstood points of the standard, including some general guidance about the new clause on conformance and compliance.



Changes in Structure - 4

1995 Original

2010 Update

Added

Annex D Reference Procedural State Model
D.1 Introduction
D.2 Procedural States
D.3 Procedural Commands
D.4 Using Collapsed or Expanded Versions of the Reference Procedural State Model

Annex B
Bibliography

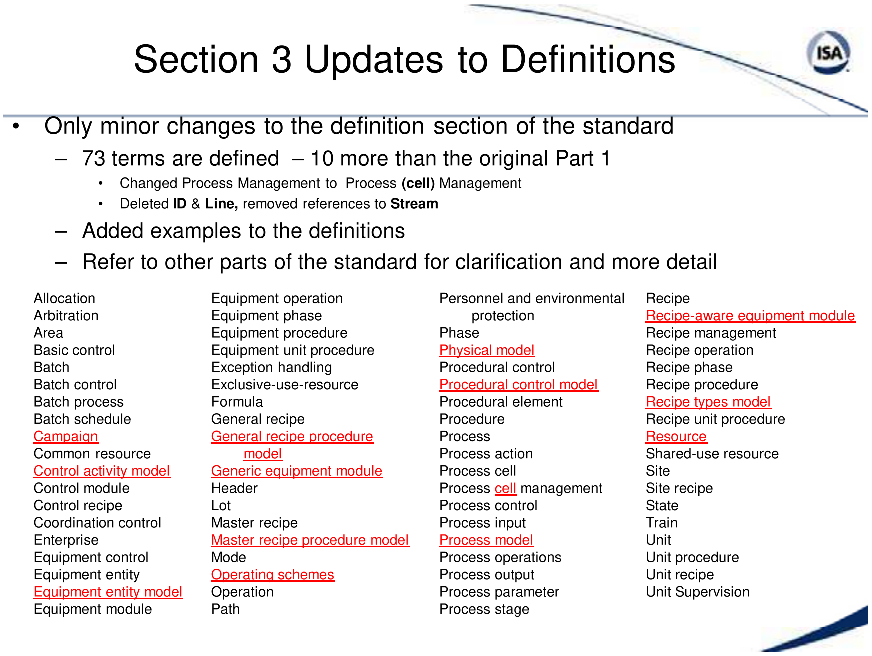
Unchanged Structure

Annex E
Bibliography

A new Annex D was added to provide a more expansive procedural state reference model. The model found in Clause 7 is considered a collapsed version of this more general model.

Bibliographical citations in the original Annex B were updated forming the new Annex E.

Get more FREE standards from Standard Sharing Group and our chats



Section 3 Updates to Definitions

- Only minor changes to the definition section of the standard
  - 73 terms are defined – 10 more than the original Part 1
    - Changed Process Management to Process **(cell)** Management
    - Deleted **ID** & **Line,** removed references to **Stream**
  - Added examples to the definitions
  - Refer to other parts of the standard for clarification and more detail

Allocation
Arbitration
Area
Basic control
Batch
Batch control
Batch process
Batch schedule
Campaign
Common resource
Control activity model
Control module
Control recipe
Coordination control
Enterprise
Equipment control
Equipment entity
Equipment entity model
Equipment module

Equipment operation
Equipment phase
Equipment procedure
Equipment unit procedure
Exception handling
Exclusive-use-resource
Formula
General recipe
General recipe procedure model
Generic equipment module
Header
Lot
Master recipe
Master recipe procedure model
Mode
Operating schemes
Operation
Path

Personnel and environmental protection
Phase
Physical model
Procedural control
Procedural control model
Procedural element
Procedure
Process
Process action
Process cell
Process cell management
Process control
Process input
Process model
Process operations
Process output
Process parameter
Process stage

Recipe
Recipe-aware equipment module
Recipe management
Recipe operation
Recipe phase
Recipe procedure
Recipe types model
Recipe unit procedure
Resource
Shared-use resource
Site
Site recipe
State
Train
Unit
Unit procedure
Unit recipe
Unit Supervision

Campaign: A collection of related batches for scheduling purposes may be called a campaign

Control activity model: a conceptual model identifying seven interdependent activities (several of which are subdivided into functions) that manage control definition, operation, and information for batch processes

Equipment entity model: a hierarchical model to logically organize the physical assets used in a batch process in combination with the control present at each level

General recipe procedure model: a conceptual model for general and site recipe procedures that structurally conforms to the process model

Generic equipment module: an equipment module which may be initiated through execution of equipment control but is not capable of being directly initiated through the execution of a recipe

Master recipe procedure model: a conceptual model for master and control recipe procedures that structurally conforms to the procedural control model

Operating Schemes: Mutually exclusive process actions executed by the same equipment module

Physical model: a hierarchical model to logically organize the physical assets of a manufacturing enterprise

Procedural control model: a hierarchical model which depicts the orchestration of procedural elements to carry out process-oriented tasks

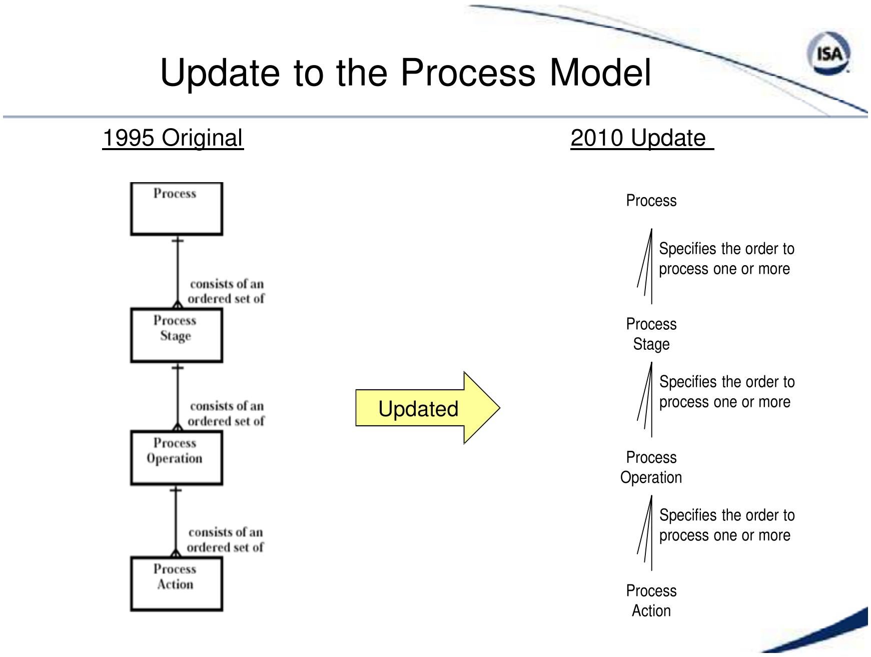
Process model: a hierarchical model which illustrates the subdivision of a batch process

Recipe-aware equipment module: an equipment module containing one or more equipment phases that is capable of being initiated directly through the execution of a recipe

Recipe types model: a conceptual model identifying the relationships between the types of recipes that are defined in an enterprise

Resource: See ANSI/ISA-95 and IEC 62264-1.

NOTE This standard focuses on equipment resources. Other resource types defined in ANSI/ISA-95 and IEC 62264-1 are for the most part not considered



Update to the Process Model

1995 Original

Process

consists of an ordered set of

Process Stage

consists of an ordered set of

Process Operation

consists of an ordered set of

Process Action

Updated

2010 Update

Process

Specifies the order to process one or more

Process Stage

Specifies the order to process one or more

Process Operation

Specifies the order to process one or more

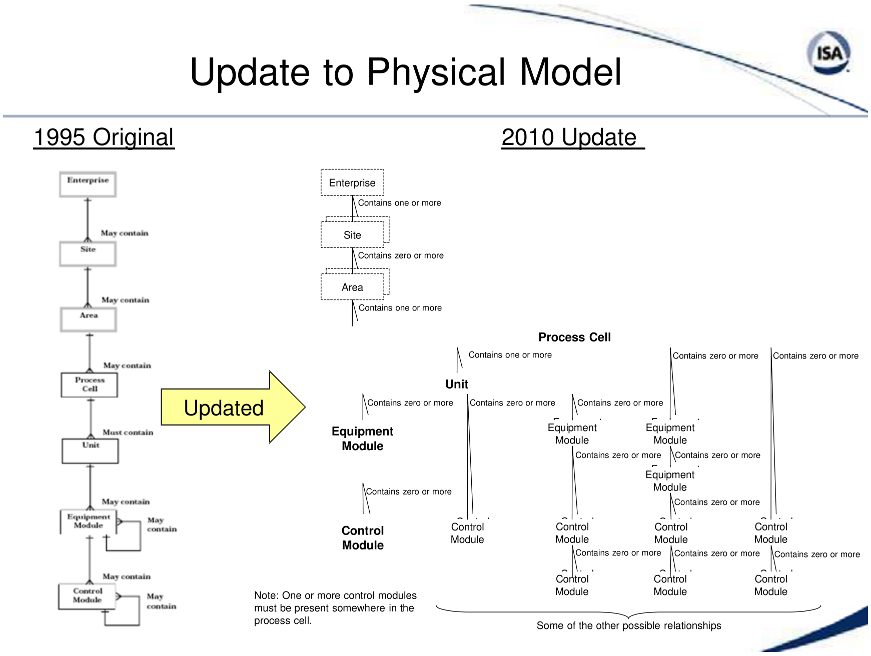
Process Action

Figure 1 (original) -> Figure 2 (update): The entity relationship diagram was replaced with a more intuitive instance diagram.

Following this same model change:

Get more FREE standards from Standard Sharing Group and our chats

- Procedural control model Figure 6 (original) -> Figure 9 (update)
- General Recipe Procedure model Figure 9 (original) -> Figure 13 (update)
- Master Recipe models Figure 10 (original -> Figure 14 (update)



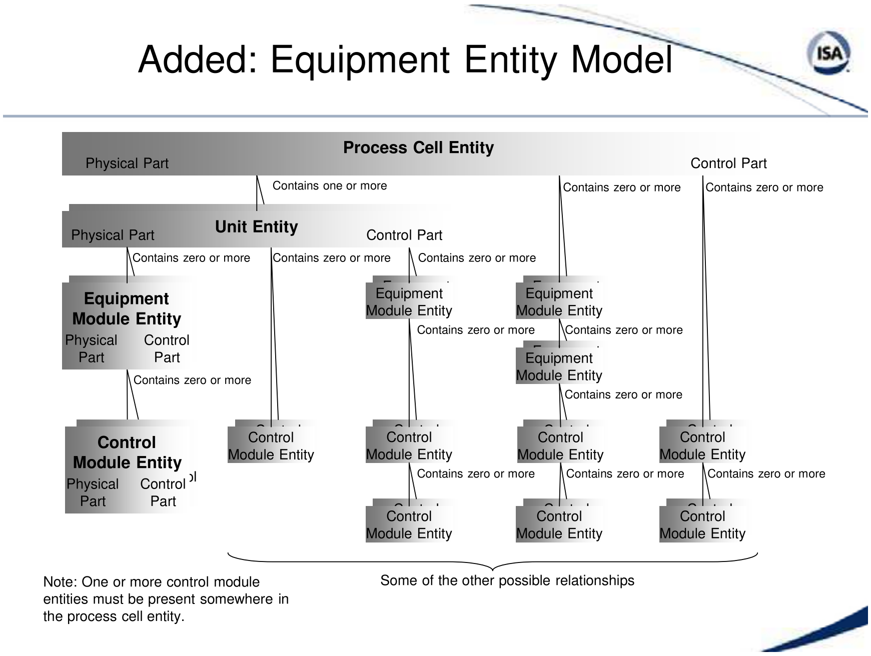
Update to Physical Model

1995 Original

2010 Update

Note: One or more control modules must be present somewhere in the process cell.

Figure 2 (original) -> Figure 3 (update): The entity relationship diagram was replaced with a more intuitive instance diagram. The figure defines equipment modules in more detail and recommends (through notes and FAQ) use of the terms compound equipment modules and compound control modules to distinguish those with subordinates at the same level in the model.



Added: Equipment Entity Model

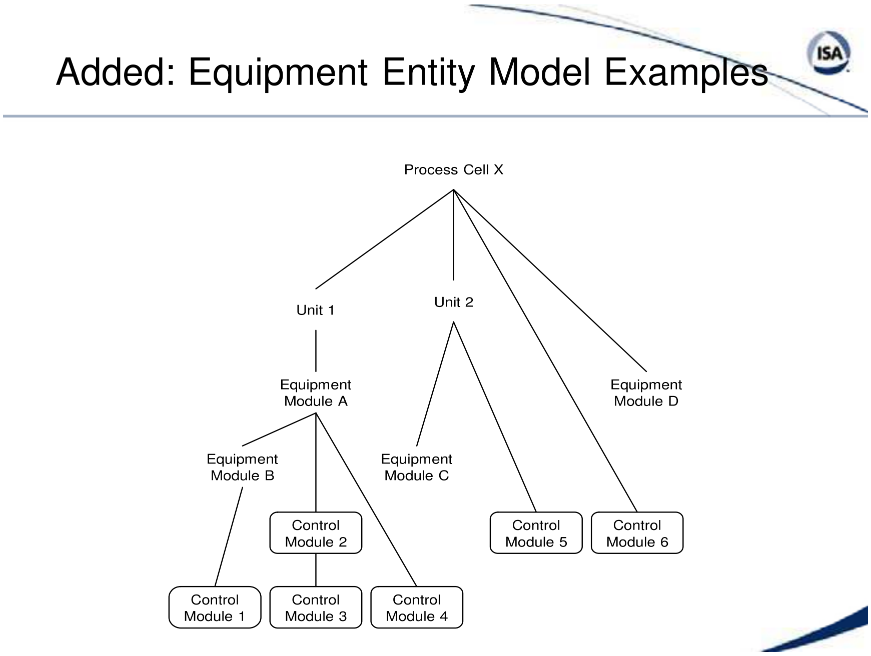
Note: One or more control module entities must be present somewhere in the process cell entity.

Some of the other possible relationships

Figure 4 (update): New instance diagram to represent the original equipment entity concept, showing that each entity has a physical part and a control part.

Get more FREE standards from Standard Sharing Group and our chats

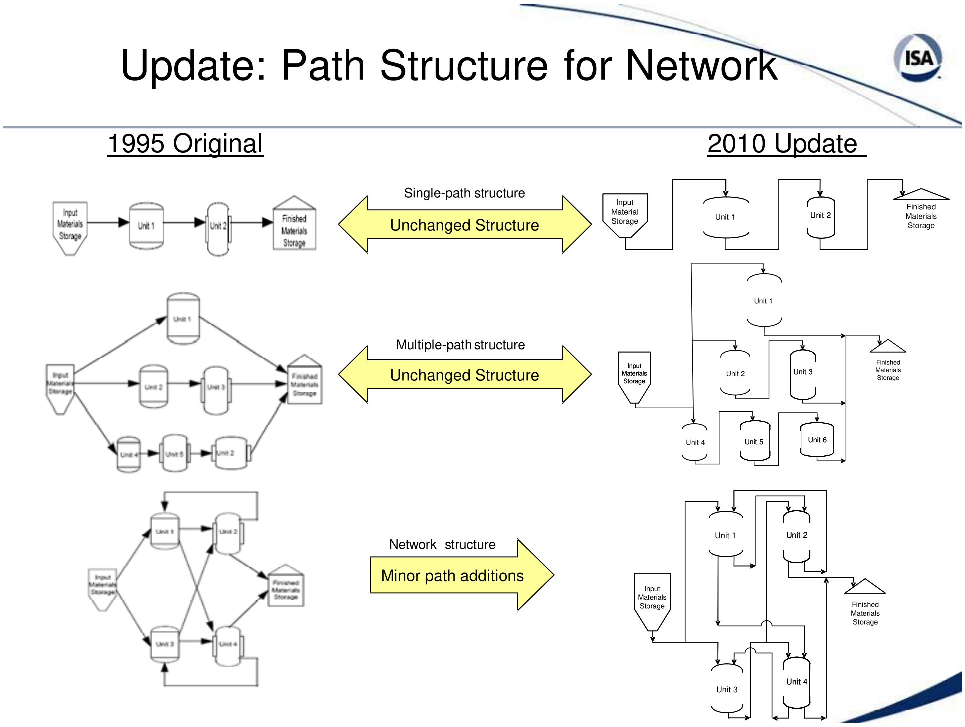
The equipment entity model hierarchically organizes equipment entities based on the lower four equipment groupings of the physical model. These groupings are created to simplify equipment operation by clearly defining the overall functions of each entity and treating it as a single larger piece of controlled equipment.



Added: Equipment Entity Model Examples

Figure 5 (update): An example is now provided in the updated standard to illustrate some expected equipment entity architectures that could be found in typical batch environment.

The added example shows that in some cases the equipment module level may be collapsed (i.e. Unit 2 contains Control Module 5 directly).



Update: Path Structure for Network

1995 Original

2010 Update

Single-path structure

Unchanged Structure

Multiple-path structure

Unchanged Structure

Network structure

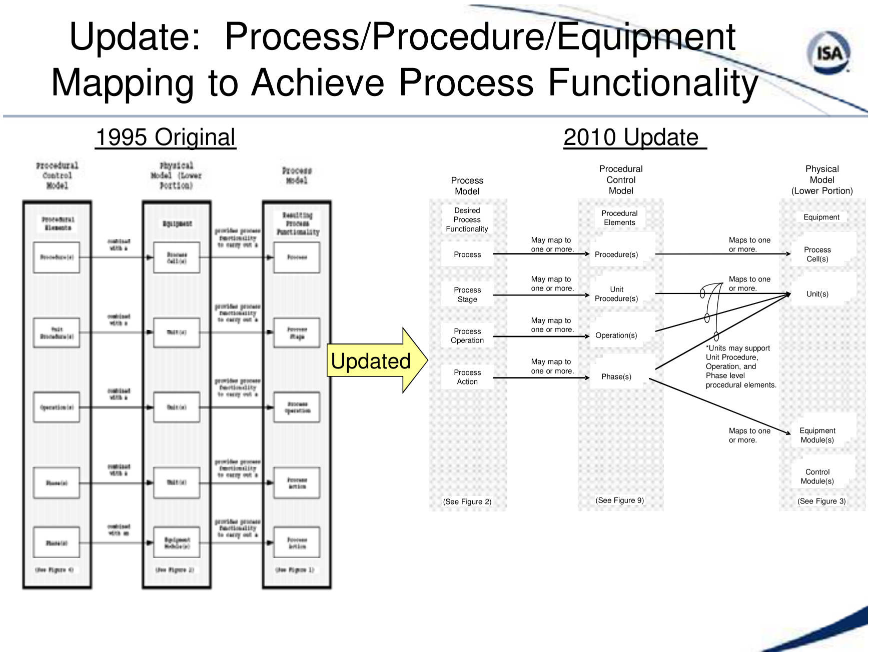
Minor path additions

Single-path structure Figure 3 (original) -> Figure 6 (update)

Get more FREE standards from Standard Sharing Group and our chats

Multiple-path structure Figure 4 (original) -> Figure 7 (update)

Network structure Figure 5 (original) -> Figure 8 (update) - Additional pathways added to network structure

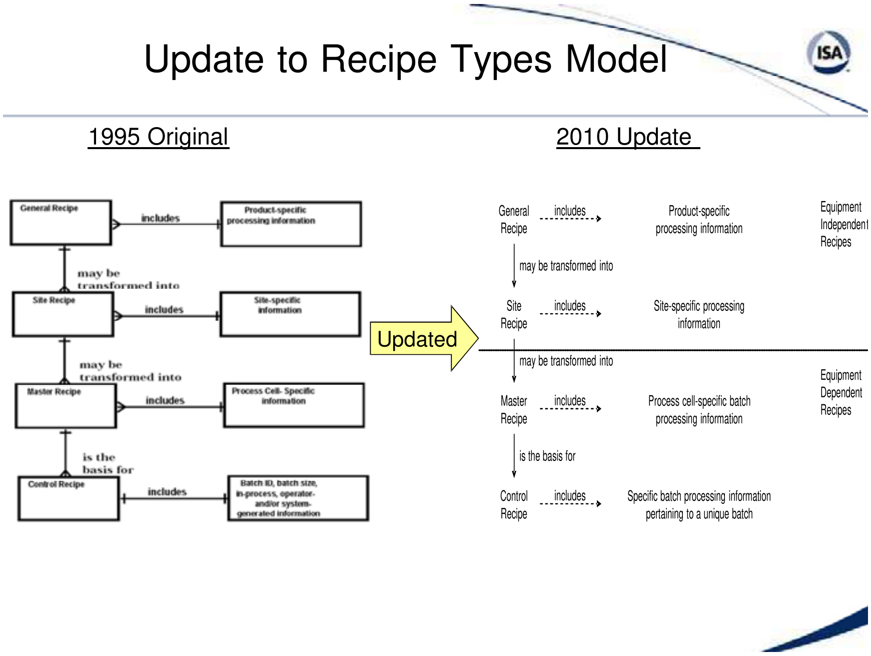


Update: Process/Procedure/Equipment Mapping to Achieve Process Functionality

1995 Original

2010 Update

Figure 7 (original) -> Figure 10 (update): The mapping of procedural control model, the physical model, and the process model was updated to show the relationships more intuitively.



Update to Recipe Types Model

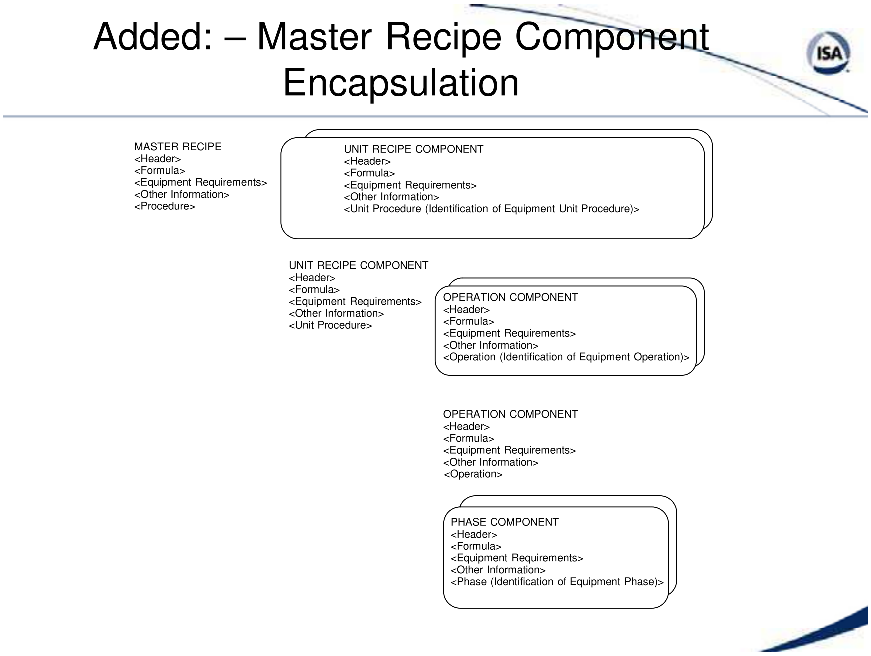
1995 Original

2010 Update

General Recipe
includes
Product-specific processing information
may be transformed into
Site Recipe
includes
Site-specific information
may be transformed into
Master Recipe
includes
Process Cell- Specific information
is the basis for
Control Recipe
includes
Batch ID, batch size, in-process, operator- and/or system-generated information
Updated
General Recipe
includes
Product-specific processing information
Equipment Independent Recipes
may be transformed into
Site Recipe
includes
Site-specific processing information
may be transformed into
Master Recipe
includes
Process cell-specific batch processing information
Equipment Dependent Recipes
is the basis for
Control Recipe
includes
Specific batch processing information pertaining to a unique batch

Figure 8 (original) -> Figure 11 (update): Updated to show equipment independence and dependence. Also updated general description for control recipe inclusion.

Procedural element relationships in the site recipe and master recipe. Figure 11 (original) = Figure 15 (update) remained unchanged



Added: – Master Recipe Component Encapsulation

MASTER RECIPE
<Header>
<Formula>
<Equipment Requirements>
<Other Information>
<Procedure>

UNIT RECIPE COMPONENT
<Header>
<Formula>
<Equipment Requirements>
<Other Information>
<Unit Procedure (Identification of Equipment Unit Procedure)>

UNIT RECIPE COMPONENT
<Header>
<Formula>
<Equipment Requirements>
<Other Information>
<Unit Procedure>

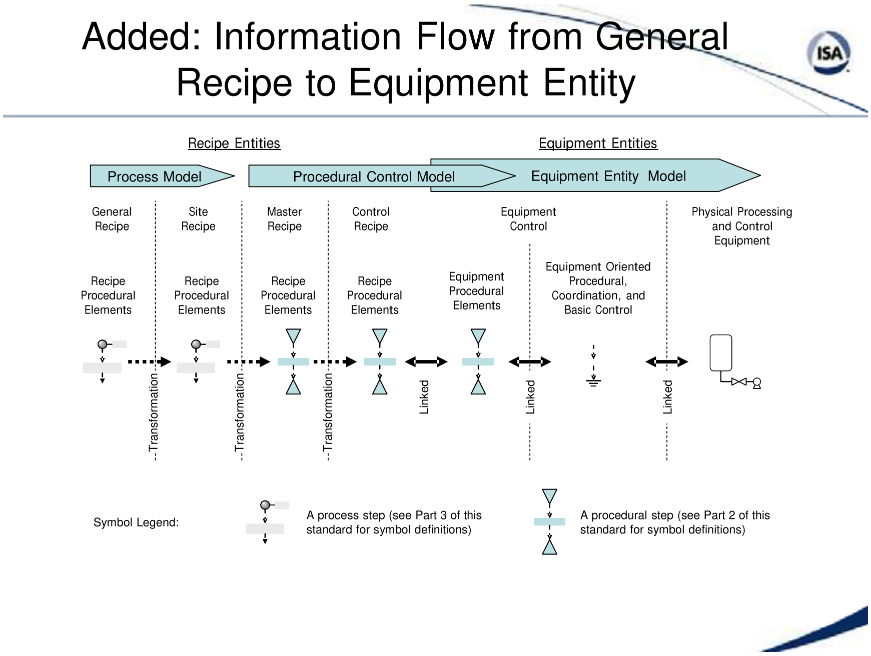
OPERATION COMPONENT
<Header>
<Formula>
<Equipment Requirements>
<Other Information>
<Operation (Identification of Equipment Operation)>

OPERATION COMPONENT
<Header>
<Formula>
<Equipment Requirements>
<Other Information>
<Operation>

PHASE COMPONENT
<Header>
<Formula>
<Equipment Requirements>
<Other Information>
<Phase (Identification of Equipment Phase)>

Figure 12 (update): This added figure illustrates how a master recipe could consist of encapsulated recipe components each containing the recipe procedural element appropriate for the level in the hierarchy and the other four defined types of associated recipe information. It shows two types of procedural relationships: 1) recipe components as recipe procedural elements that identify the related equipment procedural elements (rectangle with rounded corners) and 2) recipe components that define the procedure as the ordered set of subordinate recipe procedural elements (rectangle with square corners).

Get more FREE standards from Standard Sharing Group and our chats



Added: Information Flow from General Recipe to Equipment Entity

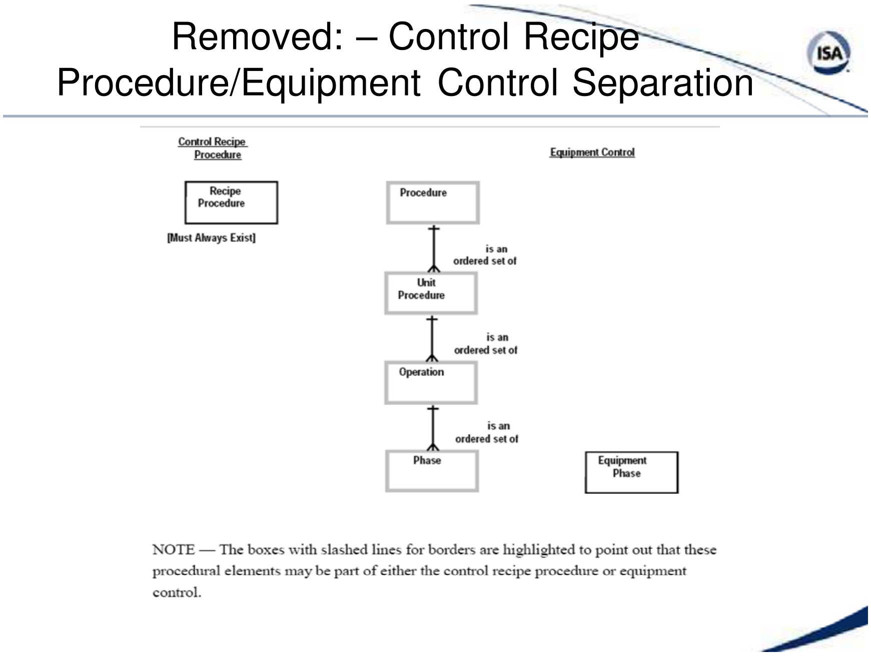
Recipe Entities
Equipment Entities
Process Model
Procedural Control Model
Equipment Entity Model
General Recipe
Site Recipe
Master Recipe
Control Recipe
Equipment Control
Physical Processing and Control Equipment
Recipe Procedural Elements
Recipe Procedural Elements
Recipe Procedural Elements
Recipe Procedural Elements
Equipment Procedural Elements
Equipment Oriented Procedural, Coordination, and Basic Control
Transformation
Transformation
Transformation
Linked
Linked
Linked

Symbol Legend:

A process step (see Part 3 of this standard for symbol definitions)

A procedural step (see Part 2 of this standard for symbol definitions)

Figure 15 (update): This added figure shows the flow of information from a general recipe to an actual equipment entity directed by a control recipe. It illustrates the abstract nature of the general, site and master recipes and their recipe procedural elements. The flow of information between recipe types is a transformation. For the control recipe to the equipment entity, and within the equipment entity, less abstract and more physical linkage should occur in order for the physical equipment to operate.



Removed: – Control Recipe Procedure/Equipment Control Separation

Control Recipe Procedure

Equipment Control

Recipe Procedure

[Must Always Exist]

Procedure

is an ordered set of

Unit Procedure

is an ordered set of

Operation

is an ordered set of

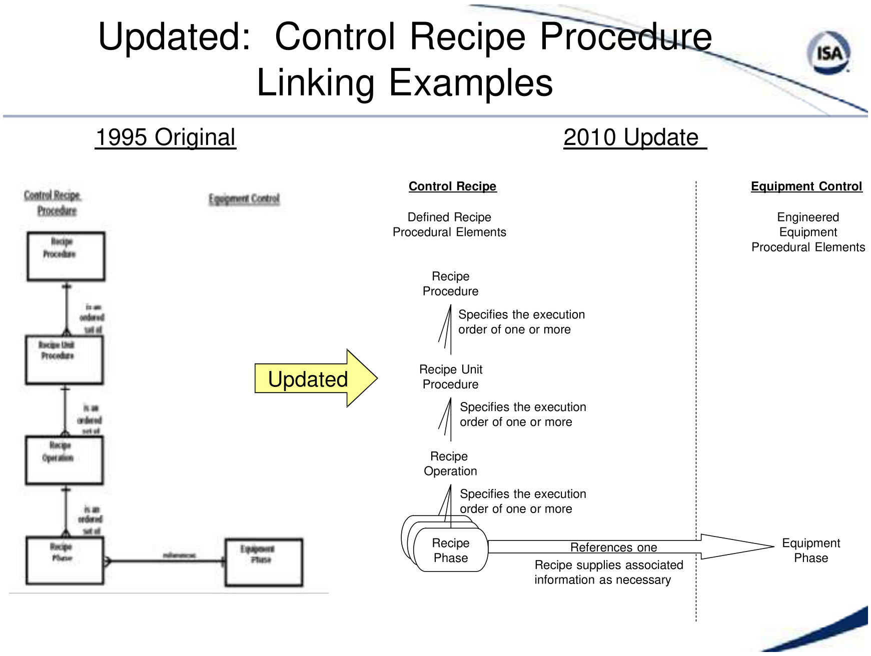
Phase

Equipment Phase

NOTE — The boxes with slashed lines for borders are highlighted to point out that these procedural elements may be part of either the control recipe procedure or equipment control.

Figure 12 (original): Did not contribute to an overall understanding of the standard. Part 1 does not concern itself with a procedural hierarchy in equipment control so this figure was removed.

Get more FREE standards from Standard Sharing Group and our chats



Updated: Control Recipe Procedure Linking Examples

1995 Original

2010 Update

Figure 13 (original) ->Figure 16 (update) Control recipe procedure referencing equipment procedural elements at the phase level. Updated to clarify purpose and details of each procedural level linking point to an equipment entity

Also modified to this structure:

- Figure 14 (original) -> Figure 18 (update): Control recipe defined without phase level recipe procedural elements
- Figure 15 (original) -> Figure 19 (update): Control recipe defined without operation level recipe procedural elements
- Figure 16 (original) -> Figure 20 (update): Control recipe defined without unit procedure level recipe procedural elements

Added: Referencing Equipment Procedural Elements at Different Levels

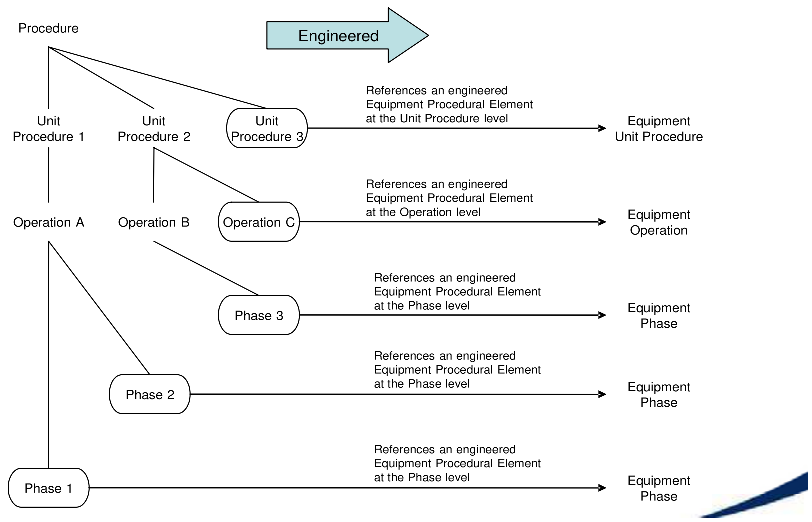

Procedure
Engineered
Unit Procedure 1
Unit Procedure 2
Unit Procedure 3
References an engineered Equipment Procedural Element at the Unit Procedure level
Equipment Unit Procedure
Operation A
Operation B
Operation C
References an engineered Equipment Procedural Element at the Operation level
Equipment Operation
Phase 3
References an engineered Equipment Procedural Element at the Phase level
Equipment Phase
Phase 2
References an engineered Equipment Procedural Element at the Phase level
Equipment Phase
Phase 1
References an engineered Equipment Procedural Element at the Phase level
Equipment Phase

Figure 20 (update): This added figure shows that when the full procedural model is not used in a recipe, a control recipe may contain recipe procedural elements at different levels (e.g. unit procedure, operation and phase) in the same recipe that each link to equipment procedural elements at their various levels. This is possible within the constraints of this standard and provides a significant degree of flexibility. As can be seen, within the same control recipe procedure, recipe unit procedures, recipe operations and recipe phases can all reference equipment entities.

Get more FREE standards from Standard Sharing Group and our chats

Updated: Procedure/Equipment Procedure Collapsibility Examples

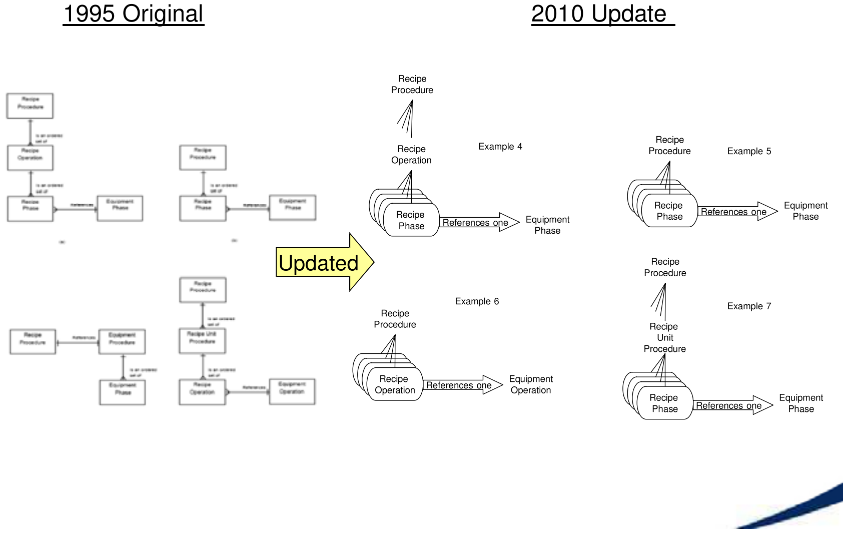


Figure 17 (original) -> Figure 21 (update): Clearer example diagrams to show collapsibility of the procedure and referencing to equipment. Simultaneous definition/selection of procedural elements and equipment entities.

Figure 20 (original) -> Figure 22 (update) has remained unchanged

Update to Example State Model

1995 Original

2010 Update

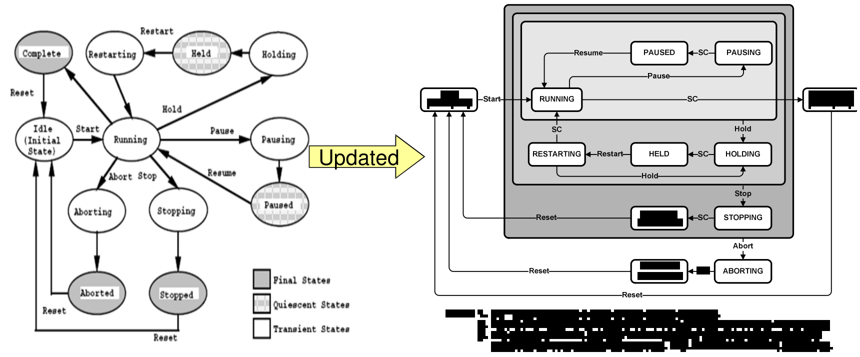
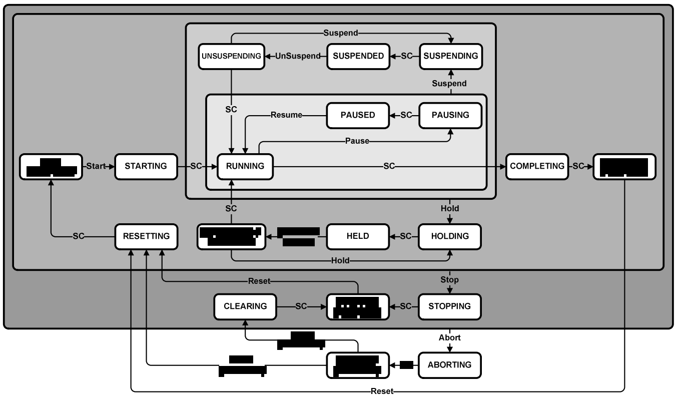


Figure 18 (original) -> Figure 23 (update): The transition diagram has been updated with a more intuitive and complete state diagram that clarifies the relationships and pathways for the procedural states example.

The following remained unchanged with the exception that Process Management was replaced with Process Cell Management:

- The control activity model, Figure 19 (original) -> Figure 24 (update)
- Recipe Management, Figure 21 (original) -> Figure 25 (update)
- Process Management is now titled Process Cell Management, Figure 22 (original) -> Figure 26 (update). Also, Manage Process Cell Resources -> Track and Allocate Process Cell Resources.
- Unit Supervision, Figure 23 (original) -> Figure 27 (update)
- Process Control, Figure 24 (original) -> Figure 28 (update). Also, Equipment Phases -> Execute Procedural Control

Second Example State Model



[redacted]

A new Annex D defines a more expansive example Procedural State Model than the one described in 7.5 to illustrate the principle of procedural states. It contains an expanded reference set of procedural states, commands, and transitions that are intended to fully meet the needs of most procedural control applications with regard to

- establishing a common understanding of and terminology for (a) the use of procedural states in exception handling and (b) communication of issues related to batch procedures and exceptions,
- consistently defining the state-related information for the recipe-equipment interface.

Get more FREE standards from Standard Sharing Group and our chats

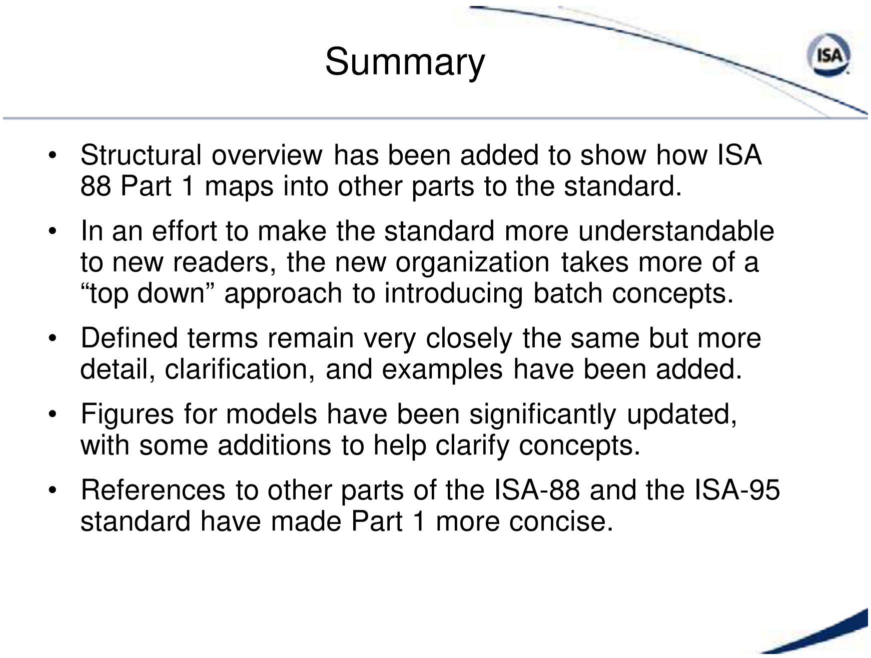
Other items

- Updated and new tables
  - Table 1 — Example modes (unchanged)
  - Table 2 — State descriptions in example procedural state model (new, updated from original text)
  - Table 3 – State transition matrix for example states for procedural elements (updated original Table 2)
  - Table 4 - State descriptions in example full reference procedural state model (new, Annex D)
  - Table 5 – State transition matrix for full reference procedural state model (new, Annex D)

- Other parts of the standard are referenced for further clarification
  - ISA 88.02- Data structures and guidelines for languages - 2 references
  - ISA 88.03- General and site recipes - 5 references
  - ISA 88.04-Batch production records - 4 references

- ANSI/ISA95 and IEC 62264-1 is referenced 16 times for further clarification

- The updated standard is about 150 pages, ~50% more than the original



Summary

- Structural overview has been added to show how ISA 88 Part 1 maps into other parts to the standard.
- In an effort to make the standard more understandable to new readers, the new organization takes more of a “top down” approach to introducing batch concepts.
- Defined terms remain very closely the same but more detail, clarification, and examples have been added.
- Figures for models have been significantly updated, with some additions to help clarify concepts.
- References to other parts of the ISA-88 and the ISA-95 standard have made Part 1 more concise.

This page intentionally left blank.

Get more FREE standards from Standard Sharing Group and our chats

Annex C
(informative)

Frequently asked questions

C.1 Conformance and compliance

Question: What is the committee's intention regarding conformance and compliance in the updated standard?

Answer:

The update committee felt that some guidance is needed relative to conformance and compliance of this part of the standard. The standard consists of models that are sufficiently abstract to apply to a wide variation of batch control implementations. Unfortunately, as a models and terminology standard with abstract concepts, examples of correct implementation and exact test criteria are difficult to define. Part 1 is intentionally abstract to allow for innovation where the committee agreed there are probably many good solutions or approaches. Yet, this part of the standard does clearly define a set of models, and terms associated with the models, which is a foundation for common understanding.

For implementation tools and specifications claiming to be compliant with this part of the standard, the committee expects that the models and terminology shall be followed and any deviations including those allowances for innovation will be clearly identified. Similarly, for a system or application to conform to this part of the standard, the committee would expect that the terms shall be used appropriately, the model hierarchies would be incorporated, and the relationships of batch elements maintained. Any deviations should be clearly defined and documented.

C.2 Multiple batches through units

Question: When a unit must continuously process material while transitioning from one batch to another, how do you manage the transition without simultaneously having multiple recipes active in the same unit?

Answer 1:

A technique using a First In / First Out (FIFO) assumption that all the material in the first batch into a unit is also the first to leave the vessel.

EXAMPLE A distillation column within a process cell.

NOTE Use of this method results in the material in two or more batches being co-mingled. This technique should only be used where co-mingling is acceptable and where physical separation of material in different batches is not a business or process requirement.

Answer 2:

A continuous transition between batches where the leading edge of a batch is used as the boundary between two batches.

EXAMPLE A textile range using recipe driven batch control to set control settings for different products.

EXAMPLE A continuous chemical plant using recipes to change grades with minimal off-spec products.

NOTE As the later batch's leading edge moves through the equipment, control settings can be changed to control the second batch in the equipment differently than the first or record keeping functions may associate data from different measurement points with the different batches. The key here is the unit is never idled as it transitions from one batch to another

NOTE Depending upon the nature of the material co-mingling between batches may occur.

Answer 3:

Impose a “transition batch” between the two primary batches to manage the transition.

EXAMPLE Same as above except for the possibly more precise control of the transition process itself and the possibility of sequester material that may need to be withheld from either of the batches due to mixing or other transition problems

C.3 Further description of basic control

Question: What characteristics of basic control separate it from procedural control?

Answer: In contrast with procedural control which is also sequential, there are two defining characteristics that separate them and a purpose that is entirely different.

— First, basic control is always present. In a steady state condition it may be quiescent[1] but is never inactive and is always open to command and may be open to recognition of a triggering condition. By contrast a procedure, such as a phase, has, by definition, a beginning and an end. Before it is activated it is totally inactive and does not have the capability to recognize any commands or triggering condition. It should also be capable of ending and becoming inactive.

— Second, basic control is defined implicitly in its design. Its various states and its response to commands and triggering conditions are implicit in the definition of its equipment and the state transitions that move it from one state to another. It is defined in terms of what it can do to or with equipment and the commands and triggering conditions that can affect the way it behaves. Its functionality in automatic mode is engineered and defined by an unchanging algorithm. All equipment and other subordinates within a control module are commanded only by the functionality designed and engineered into that single overall module. Procedural control, by contrast, is explicit and is defined in terms of the sequence of commands its designer wants to be issued while it is active in order to complete a specific task. In addition, many different procedures can (and do in practice) command the same subordinate (usually basic) control.

Get more FREE standards from Standard Sharing Group and our chats

— The difference in purpose is also clear. The purpose of basic control is to control equipment directly to set equipment to a commanded state, a process condition to a commanded value or to cycle equipment through a finite and essentially unvarying sequence based on some triggering condition. It operates based totally on command from a procedure or procedures (manual or automated). The purpose of procedural control is to command basic control in a planned sequence designed to carry out a defined task.

In essence, basic control is basic only in the sense that it is the base line or fundamental control that should be present in order to actually control equipment. It may be complex or simple, but it is the basic kind of control that should exist if equipment is to be controlled at all. Procedural control is in addition to basic control and actually commands it to the state(s) required to carry out a the desired procedure, such as making a batch of material or, in its simplest form, accomplishing a task according to an explicit sequence or *procedure*.

— Basic control is “basic” only in the sense that it is fundamental and is the essential control that sets states and conditions in the equipment or the manufacturing process regardless of the mechanism or method used to make it function i.e. automatic, manual, mechanical, etc.

- Basic control can be simple in nature but can be and often is very complex and sophisticated i.e. model reference control, multivariable control, complex recurring sequence control, etc.
- Basic control encompasses traditional stable state control including state transitions.
- Basic control encompasses regulatory control that sets a manufacturing value or condition and regulates one or more pieces of equipment in order to maintain that condition in spite of upsets and other external forcing functions.
- Basic control encompasses repetitive control actions that, once commanded to on, causes a preordained sequence of states to be activated based on a triggering condition or signal i.e. In a very simple case - a sump pump that, once turned on or commanded to be active, will actuate the pump motor when the level in the sump exceeds a given level. A more complex case might involve a more complex sequence of states that will be triggered in some order ending up back at the original state waiting for another triggering signal.
- Basic control has modes e.g. manual, automatic, etc.
- Basic control can sense condition vial equipment i.e. switches, sensors, etc. and can convert the sensed conditions into other forms such as engineering units, named conditions, etc.
- A characteristic of basic control is that it may have a mode of operation where it is continuously aware of its environment and may react to process without outside commands.
- Basic control is process and equipment oriented and is product independent. Other types of control may issue commands or set parameters to alter basic control's actions
- Components of basic control may issue commands to and set parameters in other basic control components.

C.4 Further description of equipment modules

Question: Why are there two types of equipment modules?

Answer:

- For many reasons, including the functionality typically needed for the design of common resources, it is necessary to have an equipment entity below the unit level that contains an equipment phase that can be initiated from a recipe. This was the intent of the equipment module in the original version of this standard. However this intended restriction was not clearly stated and, in practice, was found to be an undesirable constraint. In order to clarify the need in this updated standard for an equipment module that can execute an equipment phase at the behest of a recipe, while not creating an undesirable constraint, two classifications of equipment modules have been introduced.
- One type, the recipe-aware equipment module, is identical to the originally intended equipment module. It is characterized by having one or more equipment phases that can be directly initiated by the execution of a recipe. This in general means that their states, state transitions, and commands must be limited to those recognized by recipes in the same process cell, thus the name "recipe-aware" for this type of equipment module. This type of equipment module may also contain, either directly or indirectly, the subordinate control modules that it requires to affect the process.

— The second type, the generic equipment module has similar properties, but lacks the ability to be initiated directly by the execution of a recipe. In many cases, the nature of the control within such modules cannot be easily distinguished as being coordination, procedural or basic control. The only absolute constraint is that it cannot be an equipment procedural element that is capable of being initiated based on a recipe. It may be activated by commands from equipment phases or any other external source, such as an operator.

— A grouping of control modules with a higher level of control that is clearly basic control and does not have an equipment phase that is capable of being initiated based on a recipe may be either a compound control module or a generic equipment module. A similar grouping that carries out a process action using procedural control is an equipment module.

— Procedural control in either type of equipment module may affect the process through commands and parameters it sends directly or indirectly to basic control, but it does not act directly on physical equipment.

C.5 Batch manufacturing roles

Question: What are typical user roles in batch manufacturing?

Answer:

Responsibility for equipment-independent recipes, equipment-dependent recipes, and equipment entities require differing skills and expertise, corresponding to the following roles:

Get more FREE standards from Standard Sharing Group and our chats

Product Responsibility Role

This role requires detailed knowledge about the product itself and how in general it is produced.

The role requires familiarity with the product Bill Of Materials (BOM), formulas and general procedures for producing the product, but not necessarily with the actual equipment that is going to be used.

This person will be responsible for General Recipes and Site Recipes. It is often the same person responsible for product or process development.

EXAMPLE Titles may be Product Specialist, Corporate Chemist or Corporate Brewmaster.

Manufacturing Responsibility Role

This role requires manufacturing responsibility and requires detailed manufacturing process and manufacturing equipment knowledge.

This role includes translation of the product requirements in the General and Site Recipes into specific manufacturing requirements based on the existing production equipment.

This person will be responsible for Master Recipes and for specifying the general manufacturing process functions required in Equipment Procedural Control to execute their intent. It is often the same person responsible for specifying the processing equipment itself.

EXAMPLE Titles may be Process Engineer, Production Manager, Plant Brew Master

Automation Design Responsibility Role

This role requires automation design responsibility for the physical equipment and requires detailed knowledge about the specific actions the equipment is required to perform. In manually managed production engineering will generally develop all of the required operating procedures and detailed work processes to operate the equipment. If automation is provided engineering will design and deliver the automation equipment, and equipment control to automate the operating procedures and basic control requirements.

This person will be responsible for design and implementation of equipment control. It is often the same person responsible for specifying the process-connected control equipment (instrumentation and control equipment), systems, and interfaces.

EXAMPLE Titles may be Automation Engineer, Automation Specialist, Process Control Engineer.

The definition of roles is not meant to specify job titles. All roles may be filled by the same person but it is important to keep separate the different perspectives and to maintain the different details of the product and manufacturing knowledge.

Additional roles could be Production Responsibility and Maintenance Responsibility. These would have the task of running and maintaining the system – not developing or changing it. There may be other roles found in a manufacturing enterprise not listed here.

C.6 Recipe building blocks

Question: How are recipe building blocks used (as defined in Part 2 of this standard)?

Answer:

Recipe creators may build master recipe procedures using recipe building blocks. The building blocks are reusable sets of recipe procedural elements that contain encapsulated procedural elements and/or reference equipment procedural elements. This allows recipe creators to create new and maintain existing master recipes without detailed knowledge of the linkages to equipment entities required for actual operation.

The building blocks should be developed and maintained by engineers that understand the references and linkages between individual recipe procedural elements and equipment entities. In order for a recipe procedural element to reference an equipment procedural element, it should have an identification that enables the element to be correctly linked. In other cases, it should reference or include other recipe procedural elements and a specification of the execution order of those procedural elements.

When a recipe building block element is used more than once in a recipe, there may be a need to uniquely identify each occurrence of the procedural element to the operator and batch history.

It is possible for the recipe creator to work with a higher level recipe procedural element for defining the procedure and still have related lower level recipe procedural elements as part of the procedure. This could occur when the higher level recipe procedural element has been pre-configured in terms of one or more lower level recipe procedural elements. When the recipe creator invokes the use of a higher level recipe procedural element, the lower level procedural elements are carried along, even though they may be invisible to the recipe creator and may become part of the control recipe procedure.

C.7 Processing a recipe

Question: How are recipe and equipment procedural elements processed to make a batch?

Answer:

A control recipe includes the recipe components that define how to produce the desired batch. As described above, the procedural portion of the recipe components may be in terms of both recipe procedural elements and referenced equipment procedural elements. Through the three types of control (procedural, coordination, and basic control) equipment entities above the control module level have the capability of processing either of these procedural forms. As illustrated in Figure 10, the process cell equipment entity is responsible for the control capabilities needed to support procedures. If the control recipe is defined in terms of a recipe procedure that has lower level recipe unit procedural elements (as shown in Figure 16, Figure 17, and Figure 18), in processing the recipe procedure, the cell will cause the appropriate unit equipment entities to process these recipe unit procedural elements. In a similar manner, unit equipment entities will use the three types of control to process unit procedural elements. Again, depending on the recipe this may require the unit to process recipe operation procedural elements, recipe phase procedural elements, equipment operation procedural element, and/or equipment phase procedural elements. The unit may engage lower level equipment to satisfy the processing required. Eventually, the procedural definition should refer to an engineered equipment procedural element at some level in the hierarchy. Equipment control in the associated equipment entity will then execute the engineered procedural element to provide the required process functionality for the batch.

If the control recipe is defined in terms of a recipe procedure that references an engineered equipment procedure (as shown in Figure 19), the process cell equipment entity will execute its related procedural control element. Similarly, the unit and equipment module's entities could be engaged to process related equipment procedural elements. In this case, aside from the initial recipe process cell procedural element, the equipment entities are only executing engineered equipment procedural elements.

Get more FREE standards from Standard Sharing Group and our chats

C.8 Exception handling details

Question: Exception handling is a significant topic in itself. Why is there such a short clause in this standard addressing this topic?

Answer:

This Part 1 standard focuses on providing the fundamental models and terminology of batch control. The important topic of exception handling is included in that this part of the standard suggests that modular design principles and procedural states/modes can be utilized in designing exception handling procedures. There is certainly the potential need to address exception handling in more detail, but it is beyond of the scope of a standard focusing on base models and terminology. In the future, another part of this standard or technical report could be dedicated to exception handling in batch control could possibly address this topic in greater detail.

Annex D
(informative)

Reference Procedural State Model

D.1 Introduction

This annex defines a more expansive example Procedural State Model than the one described in Clause 7.5 to illustrate the principle of procedural states. It contains an expanded reference set of procedural states, commands, and transitions that are intended to fully meet the needs of most procedural control applications with regard to

- establishing a common understanding of and terminology for (a) the use of procedural states in exception handling and (b) communication of issues related to batch procedures and exceptions,
- consistently defining the state-related information for the recipe-equipment interface.

These benefits can be fully achieved by applying either the full reference model or a collapsed version of it as needed across an entire application. The example procedural state model described in Clause 7.5 and the Base State Model described in ISA-TR88.00.02 are two possible results of collapsing this reference model. It is expected that a collapsed version of the model will suffice in most cases. Extensions to the Reference Model can also be applied, but with some corresponding loss of the above benefits.

The states, commands, and transitions comprising the Reference Procedural State Model are summarized in the state transition matrix (see Table 5). The corresponding state transition diagram is shown in Figure 29. This diagram groups states together which may all respond to a common command. As illustrated, RUNNING, HELD, SUSPENDING, COMPLETING, etc. all may be commanded to STOP and a transition to the STOPPING state will be initiated.

As seen from the state transition diagram, a single execution of the process-oriented task occurs from the time the procedural element transitions from its initial waiting state (IDLE) to STARTING or RUNNING until it eventually transitions to a final waiting state that indicates either normal completion (COMPLETE) or abnormal termination (STOPPED or ABORTED). The state of the procedural element progresses through one or more acting states to carry out the process-oriented task, but in the course of handling exceptions may include one or more waiting states. Upon this procedural element reaching a final state, the higher level procedural element (if any) that initiated its execution may progress its own operating sequence to the next step.

NOTE Except where there is determined to be a specific need, it is generally recommended to inhibit the STOP and ABORT commands while in the IDLE, COMPLETE, STOPPED, and RESETTING states. This prevents activating the STOPPING, STOPPED, ABORTING, ABORTED, and CLEARING states between executions of the process-oriented task, during which time the procedural element may no longer be allocated to or viewable from a recipe.

The possible association of a unique sequence of actions with each state of an equipment procedural element is a fundamental principle of this standard, which enables complex process-oriented exception handling requirements (see Clause 7.4) to be easily and consistently dealt with through the structure of a well defined reference procedural state model. In this context, procedural state commands may initiate a normal progression to the next state, a temporary interrupt, or an abnormal termination of the procedural element, each of which initiates different state-specific actions.

The commands used to support exception handling should be applied as described in Clause 7.4 for PAUSE, HOLD, STOP, and ABORT and as follows for the SUSPEND command:

- The SUSPEND command is generally triggered automatically upon encountering a resource constraint if it is desired either to perform a controlled shutdown of affected equipment or to monitor its utilization. This is generally applied to monitor and control short stop events that are not process deviations and from which the normal running state can be automatically resumed.

NOTE Typical resource constraints are when a raw material source, product destination, utility, or other resource is temporarily unavailable.

D.2 Procedural states

The following functionality is defined for the procedural states of the full reference procedural state model. The state transition information provided is based on using the full model.

Table 4 — State descriptions in the example full reference procedural state model

State	Description
IDLE (Initial State)	Waits for the procedural element to be allocated and started, both of which may be initiated through execution of another procedural element or by other means. Upon receiving a START command while the associated permissives are satisfied, control passes to STARTING.
STARTING	Directs initial actions that should execute only once each time the process-oriented task begins. Example actions: record initial levels, reset flow totalizers, adjust alarm setpoints Upon completion, control passes to RUNNING.
RUNNING	Directs normal sequencing of the process-oriented task and handling of routine variations that can be addressed without interruption by an exception handler. Upon completion, control passes to COMPLETING.
COMPLETING	Performs any final actions that should execute only once after completion of the normal RUNNING logic. Example actions: record final tank levels or totalized flow quantities Upon completion, control passes to COMPLETE.
COMPLETE (Final State)	Waits in the final state for a RESET command after the process-oriented task has run to completion. Upon receiving an RESET command, control passes to RESETTING.

Get more FREE standards from Standard Sharing Group and our chats

State	Description
RESETTING	Prepares the procedural element and equipment for the next execution of the Process-oriented task. Example actions: reset alarm setpoints and data buffers, verify that equipment is ready for reuse Upon completion, control passes to IDLE. Note: This state always becomes active between executions of the Process-oriented task, at which time the procedural element may no longer be allocated to or viewable from a recipe. Consequently, an alternate operator interface is needed if the actions defined may require operator intervention.
PAUSING	Temporarily interrupts the RUNNING state after receiving a PAUSE command. While in the PAUSING state, the normal RUNNING logic continues to execute until reaching a defined stopping point. This could be initiated to halt execution of the normal RUNNING logic when its processing actions have reached a safe or stable stopping point, such as for a short-term stop that does not require any additional shutdown or restarting actions. Upon reaching a defined stopping point in the normal RUNNING logic, the RUNNING logic is halted and control passes to PAUSED.
PAUSED	Waits for a RESUME command before returning to RUNNING. Upon receiving a RESUME command, control passes to RUNNING and resumes normal execution immediately following the stopping point that was reached while PAUSING.
SUSPENDING	Temporarily interrupts RUNNING, PAUSING, PAUSED, or UNSUSPENDING after receiving a SUSPEND command. This could be initiated automatically to perform a controlled shutdown of affected equipment upon encountering a resource constraint, after which the normal running state can be automatically resumed. Upon completion, control passes to SUSPENDED.
SUSPENDED	Waits for an UNSUSPEND command before proceeding to the appropriate restarting state. When the associated permissives are satisfied, the UNSUSPEND command is automatically activated and control passes to UNSUSPENDING.
UNSUSPENDING	Performs any restarting actions that must be executed before returning to RUNNING after being SUSPENDED. Upon completion, control passes to RUNNING.

State	Description
HOLDING	Temporarily interrupts RUNNING, PAUSING, PAUSED, SUSPENDING, SUSPENDED, UNSUSPENDING, or UNHOLDING after receiving a HOLD command. This could be initiated to put the procedural element and equipment into a known safe state, after which the normal running state can be manually resumed. Upon completion, control passes to HELD.
HELD	Waits for an UNHOLD command before proceeding to the appropriate restarting state. Upon receiving an UNHOLD command while the associated permissives are satisfied, control passes to UNHOLDING.
UNHOLDING or RESTARTING	Performs any restarting actions that must be executed before returning to RUNNING after being HELD. Upon completion, control passes to RUNNING. NOTE 2 – The state names UNHOLDING and RESTARTING may be used interchangeably.
STOPPING	Terminates IDLE, STARTING, RUNNING, PAUSING, PAUSED, SUSPENDING, SUSPENDED, UNSUSPENDING, HOLDING, HELD, UNHOLDING, COMPLETING, COMPLETE, or RESETTING after receiving a STOP command. This is usually initiated to perform a controlled normal stop, after which the current execution of the process-oriented task cannot be returned to the RUNNING state. Upon completion, control passes to STOPPED.
STOPPED (Final State)	Waits in the final state for a RESET command after the Process-oriented task has been stopped. Upon receiving an RESET command, control passes to RESETTING.
ABORTING	Abnormally terminates IDLE, STARTING, RUNNING, PAUSING, PAUSED, SUSPENDING, SUSPENDED, UNSUSPENDING, HOLDING, HELD, UNHOLDING, COMPLETING, COMPLETE, STOPPING, STOPPED, CLEARING, or RESETTING after receiving an ABORT command. This is usually initiated to perform an immediate abnormal stop, after which the current execution of the process-oriented task cannot be returned to the RUNNING state. Upon completion, control passes to ABORTED.
ABORTED (Final State only if CLEARING is not used)	Waits in the final state for either a RESET command or a CLEAR command after the Process-oriented task has been aborted. If it has a CLEARING state, control passes to CLEARING upon receiving a CLEAR command; if it does not, control passes to RESETTING upon receiving a RESET command.

Get more FREE standards from Standard Sharing Group and our chats

State	Description
CLEARING	Allows for clearing of faults that may have occurred or for clearing of material from the equipment after ABORTING. Note: This typically requires some level of manual intervention that must be acknowledged as part of its execution sequence. Upon completion, control passes to STOPPED.

D.3 Procedural commands

The following functionality is defined for the procedural state commands when the full Reference Procedural State Model is used:

- START: This command orders the procedural element to transition from the IDLE state to the STARTING state.
- RESET: This command orders the procedural element to transition from the COMPLETE, STOPPED, or ABORTED state to the RESETTING state. In the case of the ABORTED state, it is only valid for procedural elements in which CLEARING has been collapsed out of the model.
- PAUSE: This command orders the procedural element to transition from the RUNNING state to the PAUSING state.
- RESUME: This command orders a procedural element to transition from the PAUSED state to the RUNNING state.
- SUSPEND: This command orders the procedural element to transition from the RUNNING or UNSUSPENDING state to the SUSPENDING state.
- UNSUSPEND: This command orders the procedural element to transition from the SUSPENDED state to the UNSUSPENDING state.
- HOLD: This command orders the procedural element to transition from the RUNNING, SUSPENDING, SUSPENDED, UNSUSPENDING, or UNHOLDING state to the HOLDING state.
- UNHOLD or RESTART: This command orders the procedural element to transition from the HELD state to the UNHOLDING state.

  NOTE – The command names UNHOLD and RESTART may be used interchangeably
- STOP: This command orders the procedural element to transition from the IDLE, STARTING, RUNNING, COMPLETING, COMPLETE, RESETTING, SUSPENDING, SUSPENDED, UNSUSPENDING, HOLDING, HELD, or UNHOLDING state to the STOPPING state.
- ABORT: This command orders the procedural element to transition from the IDLE, STARTING, RUNNING, COMPLETING, COMPLETE, RESETTING, SUSPENDING, SUSPENDED, UNSUSPENDING, HOLDING, HELD, UNHOLDING, STOPPING, STOPPED, or CLEARING state to the ABORTING state.
- CLEAR: This command orders the procedural element to transition from the ABORTED state to the CLEARING state, but is only valid in procedural elements for which CLEARING has not been collapsed out of the model.

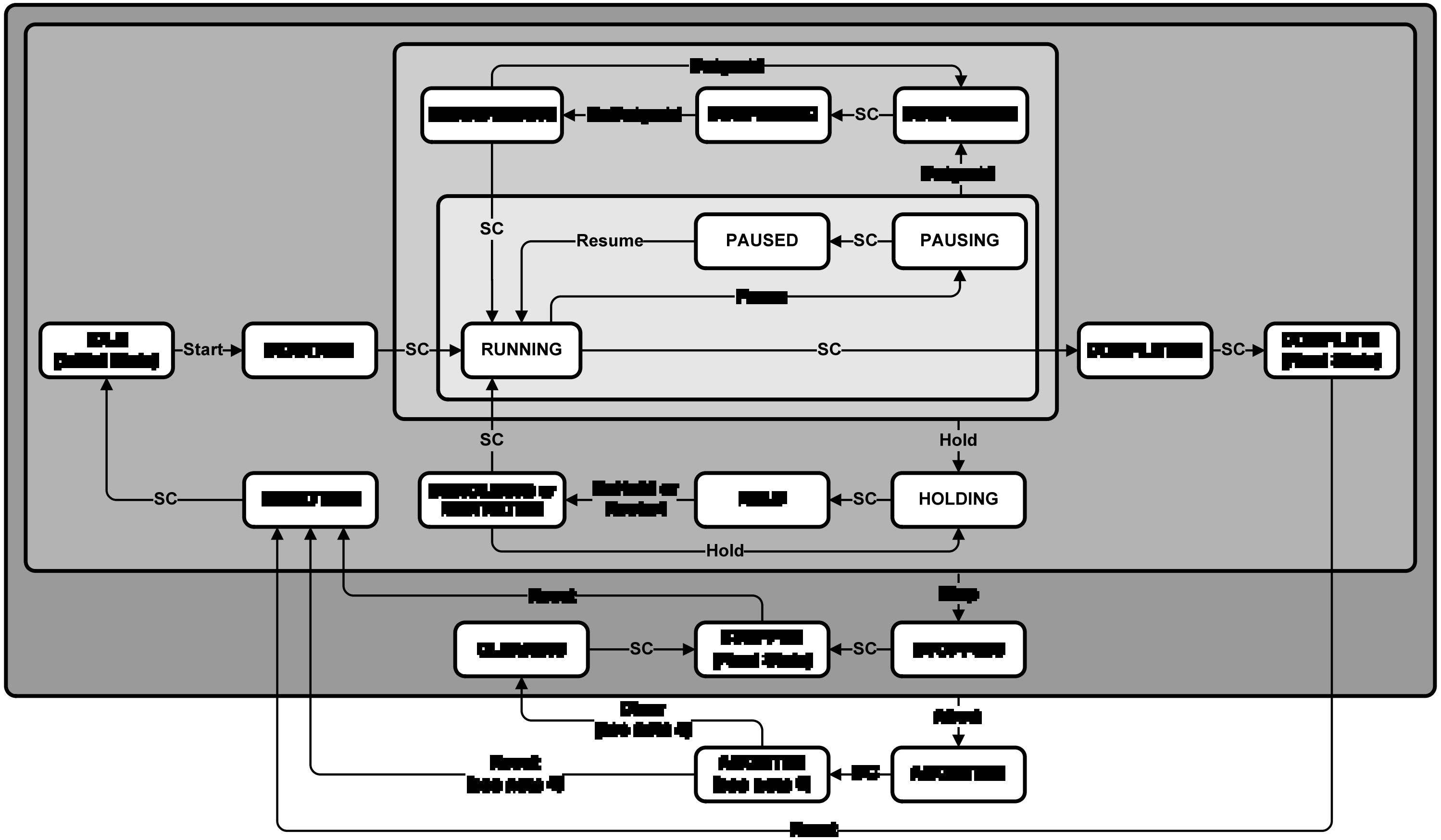
Table 5 — State transition matrix for the reference procedural state model

Current State	Transition End State when Sequence Complete	Transition End State upon receiving each Valid State Command										
		START	RESET	PAUSE	RESUME	SUSPEND	UN-SUSPEND	HOLD	UNHOLD or RESTART	STOP	ABORT	CLEAR
IDLE		STARTING								STOPPING	ABORTING	
STARTING	RUNNING									STOPPING	ABORTING	
RUNNING or EXECUTE	COMPLETING			PAUSING		SUSPENDING		HOLDING		STOPPING	ABORTING	
COMPLETING	COMPLETE									STOPPING	ABORTING	
COMPLETE			RESETTING							STOPPING	ABORTING	
RESETTING	IDLE									STOPPING	ABORTING	
PAUSING												
PAUSED					RESUME							
SUSPENDING	SUSPENDED							HOLDING		STOPPING	ABORTING	
SUSPENDED							UNSUSPENDING	HOLDING		STOPPING	ABORTING	
UNSUSPENDING	RUNNING					SUSPENDING		HOLDING		STOPPING	ABORTING	
HOLDING	HELD									STOPPING	ABORTING	
HELD									UNHOLDING or RESTARTING	STOPPING	ABORTING	
UNHOLDING or RESTARTING	RUNNING							HOLDING		STOPPING	ABORTING	
STOPPING	STOPPED										ABORTING	
STOPPED			RESETTING								ABORTING	
ABORTING	ABORTED											
ABORTED			RESETTING (see note 2)									CLEARING (see note 2)
CLEARING	STOPPED										ABORTING	

NOTE 1 The states ending with “ING” are acting states. If their logic completes normally, then a state transition to the state listed under “Transition End State when Sequence Complete” occurs. For example, if the RUNNING state completes normally, then the state automatically transitions to COMPLETE. Execution of the acting states (ending in -ING) is governed by the mode.

NOTE 2 In any procedural elements using a collapsed version of this model that does not include CLEARING, the Clear command is invalid and ABORTED is a Final State; in those that include CLEARING, Reset is invalid while in the ABORTED state.

Get more FREE standards from Standard Sharing Group and our chats



Notes: 1. SC = State Change as a result of state actions completed.
2. Actions of an equipment procedural element are generally defined by its Acting States.
3. The light, light+medium, light+medium+dark, and light+medium+dark+very dark gray boxes represent collections of states that can be preempted using the Hold, Stop, and Abort commands, respectively.
4. In any procedural elements using a collapsed version of this model that does not include CLEARING, the Clear command is invalid and ABORTED is a Final State; in those that include CLEARING, Reset is invalid while in the ABORTED state.

Figure 29 — State transition diagram for the reference procedural state model

D.4 Using collapsed or expanded versions of the Reference Procedural State Model

It may be beneficial to design recipe-equipment interfaces to support equipment procedural elements that utilize both the full reference procedural state model and collapsed versions of it. They may also support extensions of this model as needed.

The state models utilized by procedural elements may differ from one another, but they should be designed in a manner that is consistent both internally and with any propagation rules utilized for the application. Except in generic equipment modules (where such names are usually function-specific), the states, commands, and transitions for each procedural element should be consistent with the definitions for the reference procedural state model or a collapsed version of it. Procedural element state models may include additional states, commands, and transitions, but recipe-equipment interface capabilities should be considered in such designs.

When constructing procedural elements based on a collapsed version of the Reference Procedural State Model, the following guidelines should be followed for consistency:

- The IDLE and RUNNING states, the START and RESET commands, and at least one final state will not be omitted.
- If the STARTING state is omitted, the START command will instead initiate RUNNING from the IDLE state.
- If the COMPLETING state is omitted, upon completion of RUNNING, control will pass to COMPLETE.
- If the RESETTING state is omitted but not the RESET command, the RESET command will instead initiate IDLE from the COMPLETE, STOPPED, or ABORT state.

Get more FREE standards from Standard Sharing Group and our chats

- If the SUSPEND or UNSUSPEND command or the SUSPENDED state is omitted, they will all be omitted, along with SUSPENDING and UNSUSPENDING.
- If the SUSPENDING state is omitted but not the SUSPEND command, the SUSPEND command will instead initiate SUSPENDED from the RUNNING or UNSUSPENDING state.
- If the UNSUSPENDING state is omitted but not the UNSUSPEND command, the UNSUSPEND command will instead initiate RUNNING from the SUSPENDED state.
- If the HOLD or UNHOLD command or the HELD state is omitted, they will all be omitted, along with HOLDING and UNHOLDING.
- If the HOLDING state is omitted but not the HOLD command, the HOLD command will instead initiate HELD from the RUNNING, SUSPENDING, SUSPENDED, UNSUSPENDING, or UNHOLDING state.
- If the UNHOLDING state is omitted but not the UNHOLD command, the UNHOLD command will instead initiate RUNNING from the HELD state.
- If the STOP command or the STOPPED state is omitted, then both STOP and STOPPING will be omitted, but STOPPED will also be omitted only if CLEARING is omitted.
- If the STOPPING state is omitted but not the STOP command, the STOP command will instead initiate STOPPED from the IDLE, STARTING, RUNNING, COMPLETING, COMPLETE, RESETTING, SUSPENDING, SUSPENDED, UNSUSPENDING, HOLDING, HELD, or UNHOLDING state.
- If the ABORT command or the ABORTED state is omitted, they will both be omitted, along with ABORTING, CLEAR, and CLEARING.
- If the ABORTING state is omitted but not the ABORT command, the ABORT command will instead initiate ABORTED from the IDLE, STARTING, RUNNING, COMPLETING, COMPLETE, RESETTING,

SUSPENDING, SUSPENDED, UNSUSPENDING, HOLDING, HELD, UNHOLDING, STOPPING, STOPPED, or CLEARING state.

- If the CLEAR command or the CLEARING or STOPPED state is omitted, then both CLEAR and CLEARING will be omitted. In this case, if ABORTED exists, it will be considered a final state, from which a RESET command will initiate RESETTING; otherwise, it will not be considered a final state.

This page intentionally left blank.

Get more FREE standards from Standard Sharing Group and our chats

Annex E
(informative)

Bibliography of safety references

1. ANSI/ISA-84.00.01-2004 Part 1 (IEC61511-1 Mod): Functional Safety: Safety Instrumented Systems for the Process Industry Sector - Part 1: Framework, Definitions, System, Hardware and Software Requirements, International Society of Automation, 2004.

2. ANSI/ISA-84.00.01-2004 Part 2 (IEC61511-2 Mod): Functional Safety: Safety Instrumented Systems for the Process Industry Sector - Part 2: Guidelines for the Application of ANSI/ISA-84.00.01 Part 1 (IEC61511-1 Mod) - Informative, International Society of Automation, 2004.

3. ANSI/ISA-84.00.01-2004 Part 3 (IEC61511-3 Mod): Functional Safety: Safety Instrumented Systems for the Process Industry Sector - Part 3: Guidance for the Determination of the Required Safety Integrity Levels - Informative, International Society of Automation, 2004.

4. ISA-TR84.00.02-2002 Part 1: Safety Instrumented Functions (SIF) – Safety Integrity Level (SIL) Evaluation Techniques Part 1: Introduction, International Society of Automation, 2002.

5. ISA-TR84.00.02-2002 Part 2: Safety Instrumented Functions (SIF) – Safety Integrity Level (SIL) Evaluation Techniques Part 2: Determining the SIL of a SIF via Simplified Equations, International Society of Automation, 2002.

6. ISA-TR84.00.02-2002 Part 3: Safety Instrumented Functions (SIF) – Safety Integrity Level (SIL) Evaluation Techniques Part 3: Determining the SIL of a SIF via Fault Tree Analysis, International Society of Automation, 2002.

7. ISA-TR84.00.02-2002 Part 4: Safety Instrumented Functions (SIF) – Safety Integrity Level (SIL) Evaluation Techniques Part 4: Determining the SIL of a SIF via Markov Analysis, International Society of Automation, 2002.

8. ISA-TR84.00.02-2002 Part 5: Safety Instrumented Functions (SIF) – Safety Integrity Level (SIL) Evaluation Techniques Part 5: Determining the PFD of SIS Logic Solvers via Markov Analysis, International Society of Automation, 2002.

9. ISA-TR84.00.03-2002: Guidance for Testing of Process Sector Safety Instrumented Functions (SIF) Implemented as or Within Safety Instrumented Systems (SIS), International Society of Automation, 2002.

10. ISA-TR84.00.04-2005 Part 1: Guidelines for the Implementation of ANSI/ISA-84.00.01-2004 (IEC61511 Mod), International Society of Automation, 2005.

11. ISA-TR84.00.04-2005 Part 2: Example Implementation of ANSI/ISA-84.00.01-2004 (IEC61511 Mod), International Society of Automation, 2005.

12. Guidelines for Safe Automation of Chemical Processes, Center for Chemical Process Safety, American Institute of Chemical Engineers, New York 1993.

This page intentionally left blank.

Get more FREE standards from Standard Sharing Group and our chats

Developing and promulgating sound consensus standards, recommended practices, and technical reports is one of ISA's primary goals. To achieve this goal the Standards and Practices Department relies on the technical expertise and efforts of volunteer committee members, chairmen and reviewers.

ISA is an American National Standards Institute (ANSI) accredited organization. ISA administers United States Technical Advisory Groups (USTAGs) and provides secretariat support for International Electrotechnical Commission (IEC) and International Organization for Standardization (ISO) committees that develop process measurement and control standards. To obtain additional information on the Society's standards program, please write:

Get more FREE standards from Standard Sharing Group and our chats

ISA
Attn: Standards Department
67 Alexander Drive
P.O. Box 12277
Research Triangle Park, NC 27709

ISBN: 978-1-936007-75-2