Graphical Modelling Framework development to identify the control system complexity associated with the continuous processes based on ISA-TR106.00.01-2013 and ISA-TR106.00.02-2017 for Omnicon S.A.

MARIA ISABEL GOMEZ

Degree Project Industrial Automation Engineering Program

Advisor:
Mg. Oscar Amaury Rojas

University of Cauca
Electronics and Telecommunications Engineering
Department of Electronics, Control and Instrumentation
Popayán, May 2019

María Isabel Gómez Mesa

Graphical Modelling Framework development to identify the control system complexity associated with the continuous processes based on ISA-TR106.00.01-2013 and ISA-TR106.00.02-2017 for Omnicon S.A.

Degree Project as professional intership presented to
Electronic Engineering and Telecommunications Faculty
of the University of Cauca
To obtain the title of

Industrial Automation Engineering Program

Advisor:
Mg. Oscar Amaury Rojas.

Popayán
2019

I dedicate the development of the project to all the professors that have been part of my training and professional growing as an Engineer.

Acknowledgements

First and foremost, I would like to thank God for my life and every opportunity to continually grow up as person and professional. I express my sincere gratitude to my family, for their continuous support and love throughout the years. I am also grateful to all my friends for being present at important moments in my training process as person and professional. Furthermore, I would like to thank my academic advisor, Professor Oscar Amaury Rojas for his support directing and guiding the project development process with innovative ideas, and finally I am very grateful with the Omnicon company and specially with the Engineer Juan Felipe Trujillo to let me experience a real engineering process.

Summary

This document describes the development process of a graphical modelling framework for continuous processes based on Industrial Society of Automation technical report (ISA TR) 106.00.01 and 106.00.02. This framework is intended to allow process control and automation engineers to get an overview of the complexity of the control system associated with the plant at high level without reading process narratives or control narratives. The graphical modelling framework focuses only on continuous processes and is expected to bring value in large plants and to drive consistency in each manufacturing technology globally by being based in a technical report powered by ISA. For the proper framework development, three specific objectives and a general objective were defined in order to limit the scope of the framework and helps to have excellent results on the framework development progress.

Abstract

The activity of graphical modelling in industrial process is indispensable because it allows the process engineers to understand, define, quantify, visualize, or simulate by referencing it to existing and usually commonly accepted knowledge. Nowadays the graphical modelling has been used in the industrial field for material flow or process behavior design. As soon as the application of automation technologies has been increasing in industrial process, it became necessary to adapt the activity of modeling to automation functionalities and offer a specific utility for process engineers to allow them to make decisions based on quantified information. This monograph explains the development of a graphical modeling framework intended to offer a quick overview of the control system complexity associated to continuous processes based on international society of automation standard. It does not exist a modelling approach that captures these aspects in a unified framework. The graphical modelling framework focuses only on continuous processes and is expected to bring value in the facilities of Omnicon client.

Contents

1 Introduction **3**

2 Background **7**

2.1 Graphical Modelling Framework 7

2.1.1 Business Process Model and Notation (BPMN) 8

2.1.1.1 Gateways 9

2.2 Continuous Processes 10

2.3 Continuous Control 11

2.3.1 Control Loop 11

2.3.2 Process Variables 13

2.4 ISA-TR106.00.01-2013 14

2.4.1 Physical Model 14

2.4.2 Procedure Requirement Model 15

2.4.3 Procedure Implementation Model 16

2.4.4 Control Implementation Module 17

2.4.5 Complex Control Implementation Module 18

2.4.6 Unit . . . 19

2.4.7 Procedural Automation . . . 19

2.4.7.1 Automation Style . . . 20

2.4.7.2 State Based Control . . . 21

2.4.7.3 Why State Based Control? . . . 23

2.4.7.4 Loop Based Control . . . 23

2.4.8 ISA TR106.00.02 . . . 24

2.4.9 Graphical Modelling Framework Stage . . . 26

2.5 Safety Instrumented Systems . . . 28

2.5.1 Safety Instrumented Function . . . 28

2.5.2 Safety Integrity Level . . . 29

2.5.3 Interlocks . . . 30

3 Development Process **31**

3.1 Control System Component Development Process . . . 32

3.1.1 System . . . 32

3.1.2 Structure . . . 33

3.1.3 Physical Model Review and Conceptualization . . . 35

3.1.4 Continuous Control Functionalities Review . . . 36

3.1.5 Operator and Process States Interaction . . . 38

3.2 Complexity Component Development Process . . . 39

3.2.1 Complexity in Logical Programming . . . 39

3.2.2 Reusability and Effort 40
3.2.3 Complexity in System Operation 43
3.2.4 State-Loop Based Control and Automation Style 45

4 Graphical Modelling Framework Tools 49

4.1 Guide to document automation functionalities by using the GMF based on Omnicon client defined approach. 50
4.2 Graphical Shapes 50
4.3 GMF Example 52
4.3.1 Example development 53

5 Conclusions and Future Work 63

5.1 Conclusions 63
5.2 Future Work 64

List of Figures

2.1 Exclusive Gateway 9

2.2 Inclusive Gateway. 9

2.3 Parallel Gateway. 10

2.4 Complex Gateway. 10

2.5 Control Loop Block Diagram. 12

2.6 Physical Model 14

2.7 Procedure Requirement Model 15

2.8 Procedure Implementation Model 16

2.9 Discharge Pumps Complex Control Implementation Module, own source. 18

2.10 Distillation Tower Unit Example, own source. 19

2.11 Automated procedure lifecycle reference model, taken from [20]. . . . 24

2.12 Procedure Automation Strategy, taken from [20]. 26

2.13 Graphical Modelling Framework Stage 27

2.14 Safety Instrumented Function in BPCS 29

3.1 Path to structuring the GMF 33

3.2 Tomato Processing Plant Physical Model 35
3.3 Interlock representation 44
3.4 State Transition Diagram, own source 46

4.1 Unit Modules, own source 54
4.2 Ignition System, taken from process HMI 56
4.3 Interlocks in a P&ID, taken from process diagrams 56
4.4 Complexity Overview, own source. 61

List of Tables

2.1 Loop Types, own source. 12

3.1 Reusability and Effort Spreadsheet 40

3.2 Reusability and Effort Percentage 40

3.3 Effort Definition, own source. 41

4.1 Interlock table information,own source 57

4.2 Inputs and Outputs Recognition,own source 58

4.3 Steps and States Identification, own source 59

4.4 Data to get Reusability Percentage,own source. 60

4.5 Data to get Opperation Percentage, own source. 60

4.6 Data to get Effort Percentage, own source. 61

Chapter 1

Introduction

As soon as the industrial manufactured business has been increasing over the last years, it also has emerged the generation of technological tools to structure the process, to design new applications and to guarantee the manufacturing operational excellence. One of the most representative and effective process design tool in the manufacturing field is the process graphical modelling that consists of representing the processes involved in an industrial facility.

The graphical models in the industrial field provide various abstractions that help with the design of process flow or the equipment behavior by using graphical modeling language tool that express information or systems in a structure defined by a consistent set of rules. Even the graphical modelling language such as BMPN, UML, or Archimate offer the path to build any process, it does not exist a model in a graphical way that offers a way to model specifically automation functionalities such as automatic or control systems.

This monograph will explain the development process of a graphical modelling framework that is intended to display the complexity of the control systems associated with continuous process. This graphical model is structured in a framework in order to offer a scheme easy to use and understanding. Furthermore, the graphical modeling framework is based on an industrial society of automation (ISA) technical report so as to offer international standard concepts and generate common understanding.

With the purpose of guarantee the development of the graphical modelling framework for continuous processes based on technical report ISA 106.00.01 and 106.00.02, the following objectives were defined.

General Objective

- To develop a graphical modelling framework to identify the control system complexity associated with the continuous processes based on technical report ISA-TR106.00.01-2013 and ISA-TR106.00.02-2017 for an international client of Omnicon S.A.

Specifics

- To establish a set of graphical shapes, tools, templates and job aids for the modelling framework for continuous processes.
- To describe by a guide how to understand and use the graphical modelling framework, based on Omnicon client defined approach.
- To create an example that applies the graphical modelling framework for continuous processes using the tools developed.

Document Structure

This document has been divided into the chapters described below.

- Chapter 1 presents the **Introduction** that contains the document main topic, objectives and the document structure.
- Chapter 2 presents the **Background** about the relevant topics concerning the research, necessary to understand the graphical modelling framework development. These topics include models and terminology.
- Chapter 3 introduces the **Graphical Modelling Framework**, and explain in detail its **development process**. This chapter carries out a conceptualization of the definitions presented in chapter 2.
- Chapter 4 includes the development process of the **support tools and documentation**.
- Chapter 5 exposes the necessary process to apply the Graphical Modelling Framework in an industrial **process example**.

Chapter 2

Background

This chapter presents the background of the relevant topics concerning to the necessary knowledge to understand the graphical modeling framework development process.

First, a context is given about a graphical modelling framework. Second, it displays a brief explanation about the continuous processes and the continuous control systems. Third, it is disclosed relevant concepts and models associated with ISA TR106.00.01 so as to understand the conceptualization made in the graphical modelling framework development process.

Then it is explained the graphical modelling framework development and use stage supported by ISA TR106.00.02 concepts and models. Further, the interlock concept is introduced and finally, it is included a safety concept.

2.1 Graphical Modelling Framework

A graphical modelling framework, in general, is a conceptual structure intended to serve as a support or guide for the building of something, or for a specific goal, that expands the structure into something useful. The structure of a graphical modelling framework is showing with a diagram technique with named symbols which represent concepts and lines that connect the symbols and indicate relationships and various other graphical notation to represent constraints. [1]

The graphical modelling framework is intended to amplify the productivity of engineers, so they can address problems that are more challenging and take decisions based on already structure information.

The structure of the graphical modelling framework needs to be based on standard modelling language such as Business Process Modeling and Notation (BPMN) [2], Unified Modeling Language (UML) [3], Systems Modeling Language SysML [4].

From now on, the Graphical Modelling Framework will be named GMF.

Note: The term GMF could be referring to an eclipse platform, that provides a generative component and run time infrastructure for developing graphical editors based on the Eclipse Modeling Framework (EMF) and Graphical Editing Framework (GEF) [5].

2.1.1 Business Process Model and Notation (BPMN)

Business Process Model and Notation (BPMN) is a standard for business process modeling that provides a graphical notation for specifying business processes in a Business Process Diagram [6] based on a flow charting technique very similar to activity diagrams from Unified Modeling Language (UML).

BPMN give organizations the ability to communicate its procedures in a standard manner, the graphical notation can facilitate the understanding of the performance collaborations and business transactions between the organizations. [7].

For the graphical modelling framework, it is important to understand how the gateway component works.

2.1.1.1 Gateways

Gateways are BPMN flow elements that are used to control how sequence flows interacts as they converge and diverge within a process, gateways enable the implementation of branching, forking, merging and joining of paths in a business process diagram.

In order to define process flow behaviors, BPMN provides different types of gateways; these are defined with icons within the diamond shape of the gateway. BPMN defines seven types of gateways. For the scope of the project the following gateways needs to understanding.

- Exclusive: The exclusive gateway is used for creating alternative paths within a process flow, for example, exclusive decisions or sequence looping, based on the condition, breaks the flow into one of the two or more mutually exclusive paths. Exclusive gateways can be represented as showing in figure 2.1.

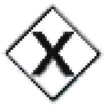


Figure 2.1: Exclusive Gateway, taken from [2]

- Inclusive: For an inclusive gateway all condition expressions are evaluated because each alternative paths is considered to be independent, any combination or the outgoing paths may be taken. Inclusive gateways can be represented as showing in figure 2.2.

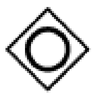


Figure 2.2: Inclusive Gateway, taken from [2]

- Parallel: A parallel gateway creates parallel paths without checking any conditions; this means that each outgoing sequence flow becomes active upon the execution of a parallel gateway, which is commonly known as a process fork.

For incoming flows, a parallel gateway will wait for all incoming flows before triggering the flow through its outgoing sequence flows, which is known as process join. Parallel gateways can be represented as showing in figure 2.3.

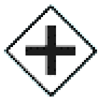


Figure 2.3: Parallel Gateway, taken from [2]

- Complex: As the name signifies, complex gateways are only used for the most complex flows in the business process. BPMN Complex Gateways can be used for splitting or merging a complicated process flow scenario. Complex gateways can be represented as showing in figure 2.4.

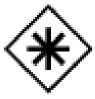


Figure 2.4: Complex Gateway, taken from [2]

2.2 Continuous Processes

It is a process characterized by a flow production used to manufacture, produce, or process materials without interruption. The continuous processes can be implemented by a control scheme where the instruments communicate continuously. In order to guarantee a quality product, it must ensure an adequate behavior of the process over the time, due to the continuous process keeps going 24 hours a day and seven days a week. The control of the continuous process behavior is made by measure and variables manipulation [8]

Generally, these processes are controlled by a distributed control system (DCS) [9], a system used for plants with a large number of control loops.

The key to quickly identify a continuous process, is because it is measured and controlled the amount of material processed per unit of time and the process is stopped only for maintenance (scheduled or not), cleaning, irregular working or abnormal situations. An example of a continuous process is petrochemical plants, that convert natural resources such as crude oil, natural gas, ores and minerals into products for a wide range of applications [10].

2.3 Continuous Control

The continuous control is characterized by a sequence of logical constant operations that measure and adjusts the controlled quantity in continuous time. In that way, the continuous control is made of control loops focus not only in the current activity but also in the ongoing effects. The variables used in the process and control system are analog type; within certain limits, the variables can take infinite values. [11].

Across the years, it was created strategies and tools, to control continuous processes. One of the main device created, was the controller, the device that is in charge of governing the process performance.

To carry out continuous control in a plant, are used control loops that relate final control elements (actuators), primary measurements elements (sensor, transmitters) with the process variables. These components are explained below.

2.3.1 Control Loop

The control loops generally cycle through the measure, decision, and actuation routine. The controlled variable is measured, and the information is sent to the controller who continually orders the actuator to look for the controlled variable to approach the desired value [12]. Figure 2.5 shows the blocks diagram of the feedback control loop.

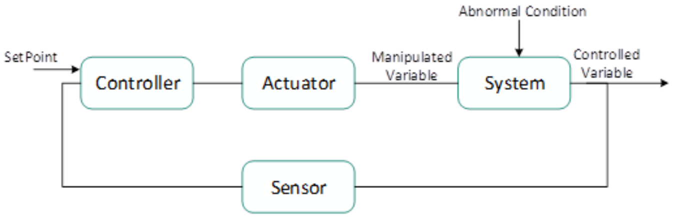


Figure 2.5: Control Loop Block Diagram, taken from [12]

The previous block diagram is not the only option available to design a control system, the following table describes the types of loops that offers some advantages important to consider.

Loop Type	Features
OL (Open Loop)	Also known as non/feedback controller. The control action does not depend on the process variable being controlled (process output). It does not use feedback to determine if the process variable has achieved the desired set point.
FB (Feedback Loop)	The control action depends on the process variable being controlled (process output). It uses feedback to evaluate if the process variable has achieved the desired set point and if not, performs required adjustment to achieve.
CS (Cascade Loop)	Arrangement of two or more control loops in series in which the output of one control loop (Primary Controller) provides the set point of another control loop (Secondary Controller)[13].
FF (Feed forward Loop)	The control action depends on a measurement of the disturbance instead on the process variable being controlled. The feed forward control loop (Secondary Controller) is normally implemented as an add-on to feedback control to take care of the major disturbance while the feedback control loop (Primary Controller) takes care of the process variable.

Table 2.1: Loop Types, own source.

The loops have different control capabilities according to the process need. For example, a control loop regardless of its type can be useful to control the quality of a product, or may be appropriate for the process control optimization.

2.3.2 Process Variables

The process variables are values gives by physical dynamics that involve the process. An example of this would be the level of a tank. The current level is called process variable, while the desired level is known as the set point.

The following types of variables are defined in order to identify which variable needs to be controlled, measure or manipulated:

- Manipulated Variables: The manipulated variables are the ones that directly influence the behavior or value of the controlled variables, and consequently in the process.

  The selection of the manipulated variable is made taking into account the one that has the greatest influence on the behavior of the controlled variable, typically is the energy flow consumed by the plant. For example in the milk pasteurization process, made by a heat exchanger, the manipulated variable is the steam applied to the exchanger [14].

- Controlled Variables: These variables follow a desired behavior over the process, which guarantee the quality of the product. Each controlled variable has its own measure instrument or signal transmitter. For example in a process of milk pasteurization, made by a heat exchanger, the controlled variable is the milk flow in the heat exchanger output [14].

- Critical Variables: These variables are also controlled variables, but they are the key variables that could affect the production process largely, these variables are generally monitored to detect deviations because they could have a higher impact on the process, so the critical variables should be prioritized and held in a stricter state of control.

2.4 ISA-TR106.00.01-2013

ISA TR106.00.01 is a technical report that provides good practices on continuous process operations by focusing on procedure automation [15]. The technical reports gather some tools that include models and concepts for the consistent use of automated procedures for continuous process operations. This technical report gathered some tools that include models and concepts for the consistent use of procedure automation in continuous process operations.

2.4.1 Physical Model

The ISA-106 physical model contains both process equipment and instrumentation. Therefore the ISA-106 physical model is shown as two objects, one representing the actual process equipment (e.g. Heat Exchanger) and the other representing the instrumentation (e.g. transmitter) which is the BPCS's connection to the process equipment. The figure 2.6 shows the physical model [15].

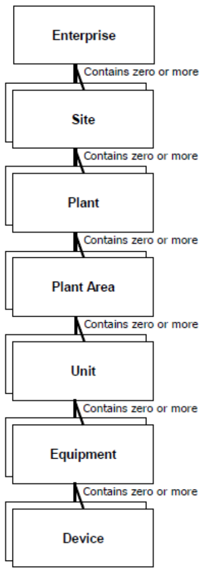


Figure 2.6: Physical Model, taken from [15]

This model include only physical assets, do not include control algorithms associated to the physical assets.

Example: The enterprise Chemistry S.A, located in the site Philadelphia contains the plant of reactions that have the plant area Sulfur production one of its units is in charge of gas release, contains the equipment Scrubber and Tail Gas demister , the tail gas demister is compose by devices such as vacuum valves, hand valves and pumps.

2.4.2 Procedure Requirement Model

Procedure requirements are the definitions of what is required to accomplish an objective. They are used as functional requirements when automated procedures are implemented. The model notation that connects model levels indicates that the upper level may contain zero or any number of lower-level objects[15].

The procedure requirement model define procedure and process control functions required to accomplish an objective by the creation of implementation modules.The figure 2.7 shows the procedure requirement model.

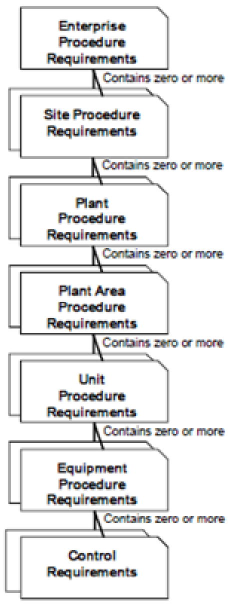


Figure 2.7: Procedure Requirement Model, taken from [15]

Example: A set of pre-defined Modes of Operation defined for communication and receiving request from above plant procedure requirement module or parallel site procedure requirement module. (States, Steps and Transitions). Please refer to subsection 2.4.7.

2.4.3 Procedure Implementation Model

The procedure implementation model is the connection between the procedure requirements model and the physical model because this model Implement procedure and process control functions defined in requirement model to be implemented in the physical model [15].

The procedure implementation model shown in figure 2.8 shows the hierarchy of implementation modules that results from implementing procedure requirements.

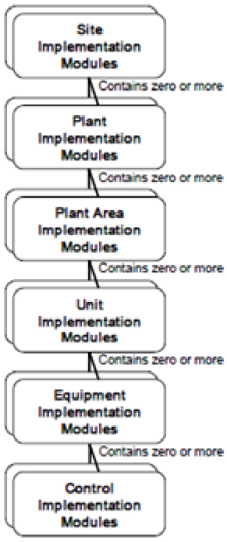


Figure 2.8: Procedure Implementation Model, taken from [15]

Example: The implementation modules run procedural (sequencing) logic and connect to physical equipment via Control implementation Modules or equipment implementation modules. Some examples of Equipment Modules are the procedures defined for plant startups or shutdowns, but they may also be used for simpler procedures such as isolating and starting up a redundant pump system, performing maintenance of a piece of equipment or performing an in-line valve performance test.

It is important to highlight that while any of the model levels may be omitted as appropriate, the unit, unit procedure requirement and unit implementation module levels are typically used as they provide the operational focus for major pieces of process equipment. Thus, it is make special relevance on unit definition: It is also important to explain that control and equipment modules in ISA 106 only contain control functions, not the physical hardware which is represented and defined in ISA − 88 . The following concepts are the relevant definitions contained in ISA TR106.00.01 that add value to the GMF and needs to be understanding. [16].

2.4.4 Control Implementation Module

It is defined as a collection of sensors, actuators, other Control Implementation Modules, and associated process equipment that, from the point of view of control, is operated as a single entity.

Control Implementation Modules drive equipment to a state (e.g. running, stopped, opened, closed, speed set), can provide information (e.g. current state or alarms conditions) and receive commands (e.g. start, stop, open, close). It may incorporate multiple inputs and outputs into a single control function.

Control Implementation Modules can be divided in two types:

- Active: Performs an action (e.g. a VFD, motors, valves).
- Passive: Provides information but does not have any action. (e.g. sensor switches, transmitters).

2.4.5 Complex Control Implementation Module

A Complex Control Implementation Module (CCM) is a group of Control Implementation Modules that have relation and junction between each other and are operated as a single module. They allow reusing the objects, standardizing and optimizing the engineering efforts. Complex Control Implementation Modules provide basic control, drive equipment to a state and may have steps.

Complex Control Implementation Modules may include more than one Control Implementation Module of different kinds such as motors, valves, PIDs, process variable transmitter, etc. An example of a Complex Control Implementation Module is a Discharge pumps CCM that is composed by 6 CMs: two pumps and two discrete valves for each pump as shown in Figure 1.6. This is considered a CCM because the 6 devices are operated as a single entity instead of operating each one as a single device. It also can have different modes of operation including: both pumps running, only one pump is running and the other one is out of service, both pumps are out of service, etc.

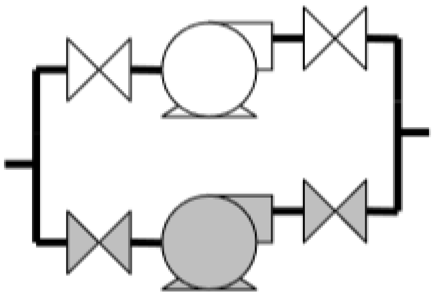


Figure 2.9: Discharge Pumps Complex Control Implementation Module, own source.

Applying these implementation modules provides consisted functions that allows the plant to development of a library of common equipment implementation modules that can be re used on a number of similar pieces of process equipment.

2.4.6 Unit

A unit consists of all equipment and devices required to perform a major processing activity on a process stream. Units operate relatively independently, although a unit may interact with other units as well e.g. when transferring material. Figure 1.7. shows an example of a distillation tower unit with the control devices that perform coordinated control functions for the distillation process.

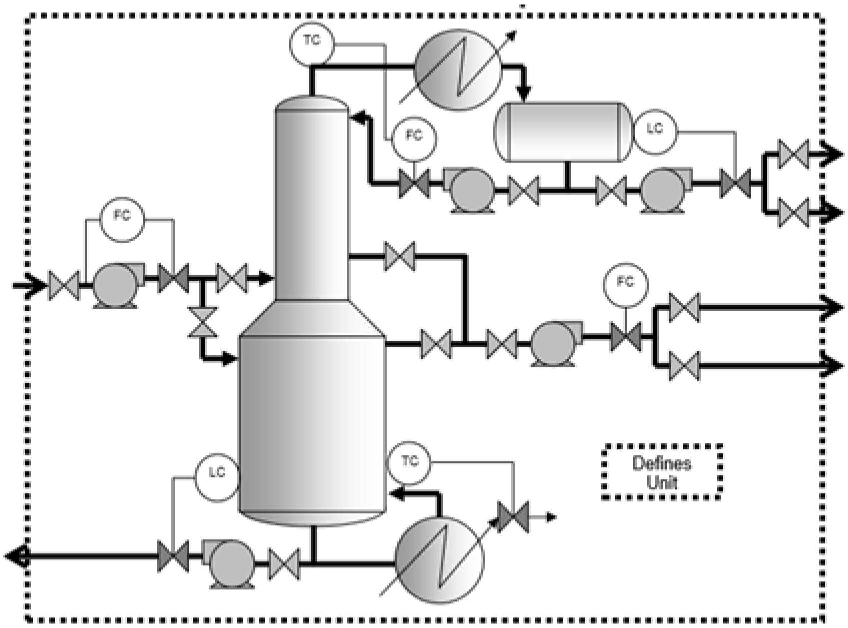


Figure 2.10: Distillation Tower Unit Example, own source.

2.4.7 Procedural Automation

ISA TR106.00.01 focus its content on offering good practices on applying procedure automation. Procedure Automation represents the implementation of a procedure on a programmable mechanical, electrical or electronic system, it means adapt process operations and strategies for incorporating automated procedures into industrial automation and basic process control systems (BPCS). The ISA TR106.00.01 models described in subsection 2.4 are defined for procedure automation of continuous process operations. Good practices in procedure automation make special relevance in control based on states and introduce the following terms:

2.4.7.1 Automation Style

When the implementation methods that result from the implementation modules are applied, create different types of control strategies and system interaction. The different types are called automation styles. An automation style relates the operator communication with the system by relating the process knowledge that resides in the automated procedures [15]. This means that in complex automated styles the automated system contains more process knowledge, making the procedure more automatically to provide consistent and efficient operations of plant processes.

The automation style provides a framework that fosters the re-use of procedure logic. ISA TR106.00.01 defines the following three common automation styles:

- Manual:The operator is responsible for the command, perform and verify work items. The operator may use the BPCS console or local indicators and actuators in the field, but there is no computerized procedure automation involved.
- Computer Assisted: Implementation modules are considered computer assisted when the operator and computer share responsibility for the command, perform and verify work items. The amount of automation used may vary.
- Fully Automated: Implementation modules are considered fully automated when the computer is responsible for the bulk of the command, perform and verify work items.

A single project may utilize more than one automation style, as each style may be most appropriate for a different part of the process or for different process conditions[15].

2.4.7.2 State Based Control

The state based control is a type of automation control design based on the principle that all process facilities operate in process states.[17] The states represent the variation between normal and abnormal conditions of the process, the variation dictates changes to the automation and control parameters of the process. When using process states, procedure automation is centered on a major piece of process equipment, usually a unit. States are defined based upon the physical conditions the process equipment passes through to ensure safe and efficient operation.[18]

Over the past years the automating processes can become very complex if no structure is used in implementation. This needed structure is best applied by introducing states as foundation.

State Based Control (SBC) is a way of automating processes equipment, applicable for discrete, continuous, batch and hybrid forms of production. It is based on the notion that every unit/module always has a state, and that the state is of primary importance to understand what the unit is doing. A unit can change state, based on process conditions or commands. An event that ends the current state and moves the unit to another is called a transition.

This state based approach is already common practice to Control Modules for decades (i.e. a valve is always opened/closed/travelling/fault) and with SBC-approach driven up to higher layers of the automation hierarchy. Every module in the hierarchy with requirements that can be represented by states, can be automated by using a procedure that follows SBC principles.

As stated, in this perception every unit ALWAYS has a state, this implicitly means also that it is never stateless. For a state to be meaningful, it must be unambiguous in its name and associated purpose.

A state is not derived from the states of the individual components that make up a unit, it is realized the other way around: a state is activated (by operator command or program transition), and all subordinate components are commanded to assume their states (set points, commands) that are required to achieve the desired state (a state forces al subordinates/child equipment).

State Based Control (SBC) can be seen as an expansion to ISA 88, which is a well-accepted standard for batch control, and as it happens is the major foundation for SBC as well. In the meantime, the industry has also recognized the need for a continuous model, and the ISA106 is being developed to fill this gap. This SBC approach is quite similar to the ISA106, and can be seen as a practical interpretation of that future standard. Where ISA TR106.00.01 is a more procedural oriented, SBC is focusing more on the process. It is completely build upon the ISA TR 106.00.01 physical model, extended for continuous operation.

This approach can coexist perfectly with a batch approach, but it also works very well in isolation to automate continuous and discrete processes that typically do not have a large variation in execution order. In these processes it is common practice to have many unit operations controlled by a single operator, who needs a high level operational state/overview each unit is in.

For the state based control application, it is necessary to define states and transitions.

- States: The states or process states are particular conditions of a piece of equipment at a specific time. Process states are defined based on the physical conditions the process equipment passes through to ensure safe and efficient operation. Process states can represent steady state conditions (e.g. Not Ready, Ready and Producing) while others represent transitory states (e.g. Preparing, Filling, Heating and Shutting Down). Transitory states are used to transition the process from one steady state to another.

- Transition: Transitions limit the possible movement between states, i.e. contain the conditions that end the current state and move to the following. Transitions are typically divided into 2 types, the manual transition that triggered by the operator through the user interface and the automatic transition that Triggered by the system based on one or more monitored values (e.g. calculations, instrument values, timers, confirmation signals, etc.).

  Some examples of transitions are manual commands, a signal that indicates the level of a tank reaches a desired set point, a motor feedback signal, a reached preparation time.

2.4.7.3 Why State Based Control?

State based control is credited for many improvements in automation and production environments, in both engineering and operational productivity. Some of today's challenges in automation:

- Repeatable, optimized startup, run shutdown of plants.
- Abnormal situation management.
- Intelligent alarming.
- Automatic (condition based) enabling/disabling of equipment.
- Automatic swapping to redundant equipment
- Automatic sequencing of (complete) plant startup
- Increase operator span of control
- Put more equipment under control of same people
- Provide better info, so more equipment is manageable

2.4.7.4 Loop Based Control

The fact of establishing states in a procedure as already described, allows the system to decide and execute commands by itself. In the case of loop-based control, when no states are defined by an implementation model then the operator needs to find out the states and has the responsibility to execute commands and provide information about the next states.

The loop-based control is the type of control design that has been applied in most of the facilities in the automation industry field.

2.4.8 ISA TR106.00.02

ISA TR106.00.02 is a technical report that describes work processes involved with automating procedures, monitor and control continuous processes [19]. This technical report is written for new process facilities as well as control upgrades to existing facilities.

ISA TR106.00.02 provides information regarding the work processes involved in the life cycle of automated procedures for continuous process operations.

The definition of the GMF stage was based on the automated procedure life cycle reference model showed in figure 2.11. In this model, the rounded rectangles represent the work processes, or their deliverable, that are directly related to the life cycle of automated procedures [20].

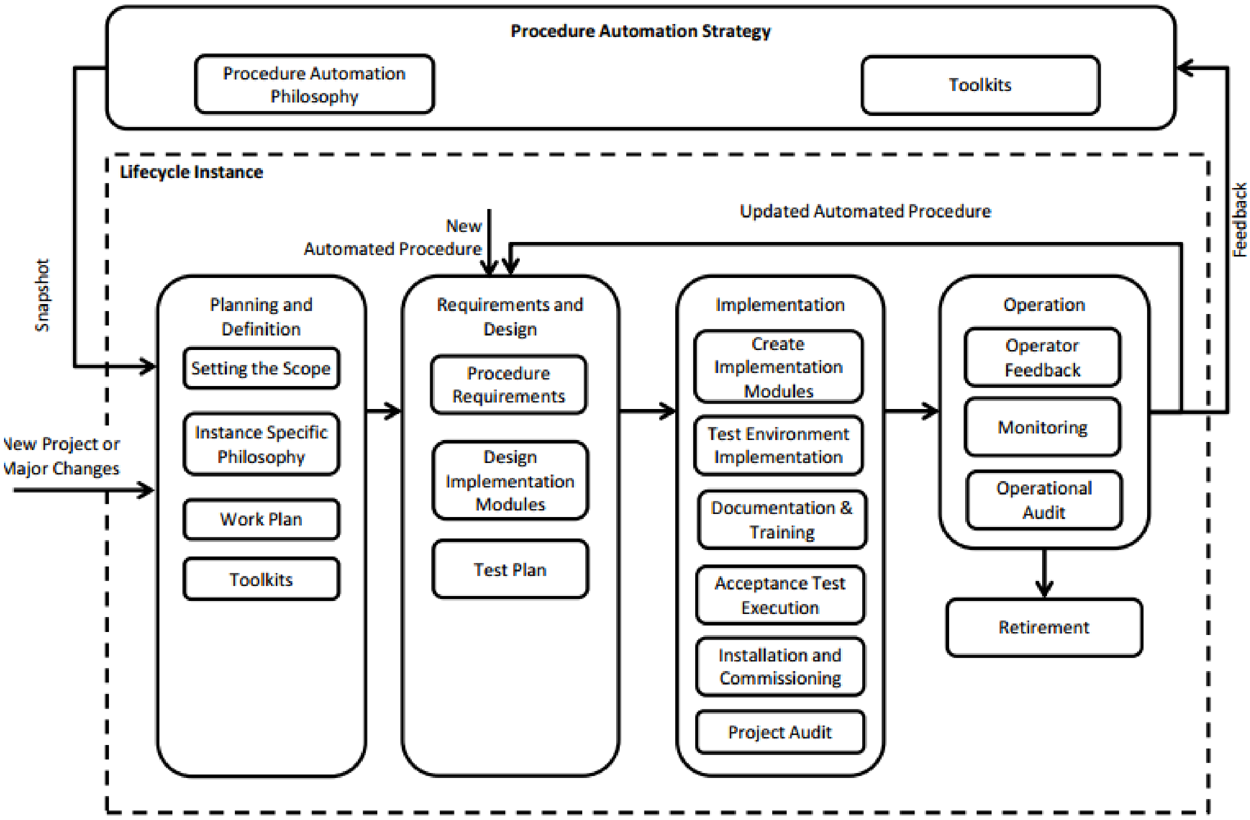


Figure 2.11: Automated procedure lifecycle reference model, taken from [20].

The automated procedure lifecycle assumes that it exists a procedure automation strategy that contains a procedure automation philosophy and toolkits to develop them.

The lifecycle instance needs to be developed based on procedure automation strategy, the steps to accomplish the lifecycle instance of a project relating to control system application in continuous processes are:

Note: The following steps were analyzed with a GAMP [20] environment based on an ISA TR106.00.02 webinar suggested approach.[21]

- Planning and Definition: This work process defines the scope, deliverable and technical approach to be used for the life cycle instance. The initial scope of the life cycle instance is set by the project or operations environment the life cycle instance exists within. In all instances, the activities are evaluated to decide if they are required for the specific instance and, if required, how extensively they should be performed. In the GAMP environment, this is how to define user requirement specification **(URS)**.

- Requirements and Design: The requirements and design work process result in the creation of the procedure requirements model and procedure implementation model for the life cycle instance. These models are used to create the implementation modules, training, and documentation in the subsequent implementation work process. In the GAMP environment, this is how to define functional requirement specification **(FRS)**.

- Implementation: The implementation work process includes activities for creating the implementation modules, testing the automated procedures, creating documentation, training material and the installation and commissioning of the automated procedures.

  In the GAMP environment, this is how do we convert what we define as FRS into software design specification **(SDS)**.

- Operation: During the operational period of a plant's life cycle, automated procedures are used by the operations staff. Concurrent to this use, a monitoring and assessment activity and an operational audit activity is performed in order to maintain and improve the automated procedures and to ensure they fulfill requirements.

There is no definition in GAMP environment for operation, this step is related to the analysis in system operation.

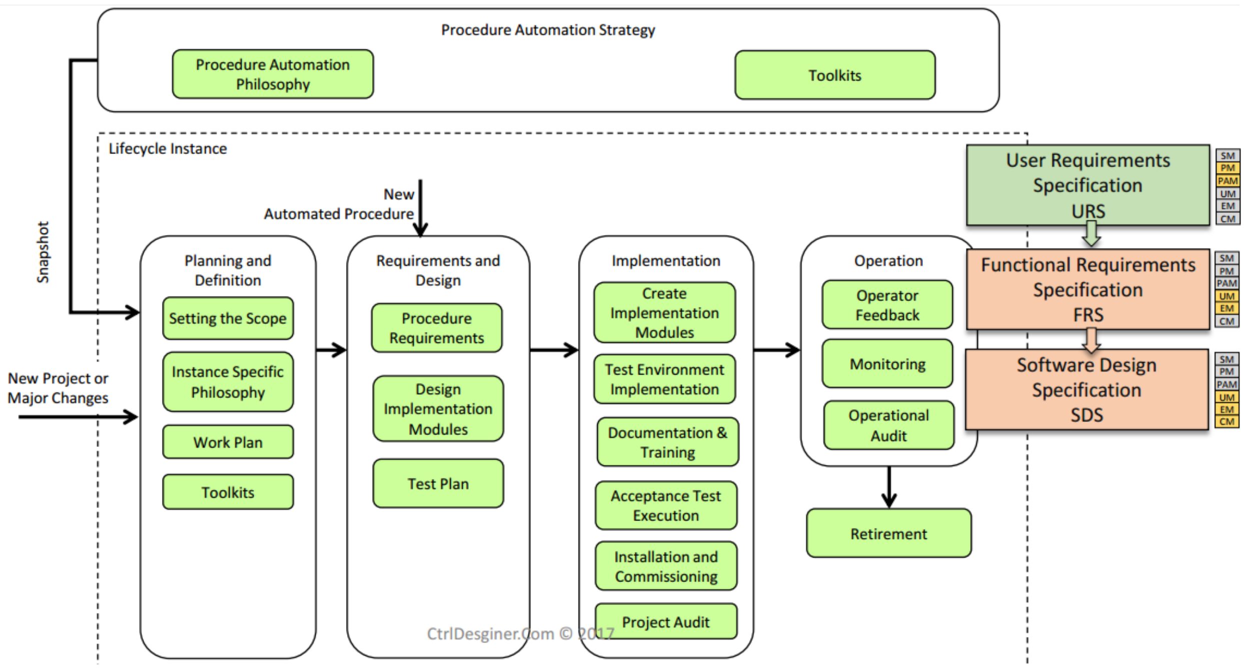


Figure 2.12: Procedure Automation Strategy, taken from [20].

It can be notice that ISA TR 106.00.01 propose to establish the user requirement specification (URS), in the plant module and plant area module level and for functional requirements specification (FRS) it is proposed to establish in unit module and equipment module level and finally the software design specification (SDS) that is proposed to be established in the unit module, equipment module and control module level.

2.4.9 Graphical Modelling Framework Stage

Based on the previous assertions, figure 2.12 was defined in order to establish the pre-requirements to build the GMF, and to determine the GMF stage. The figure

?? establish an arrangement where GAMP, ISA TR106.00.01, and ISA TR106.00.02 concepts are involved.

This arrangement represents the stages that need to be assumed to model and design the automation functionalities. In order to understand how the concepts are met, follow the explanation right below: Chiefly, the procedure automation strategy is composed of the continuous control and procedure automation philosophy; these philosophies are intended to provide consistency of automated procedures between lifecycle instances.

The procedure automation philosophy differs from the continuous control philosophy as long as the procedure is focused on procedural automation such as the behavior from an operator perspective, operator interface, operator-system interactions, or automation styles used. The continuous control philosophy provides the overall continuous control conceptual design based on the PFD with all principle control loops required to make effective use of the facility and manage expected quality, throughput and yield variations in the process while ensuring that the quality and volume of the finished product is within a target range.

Therefore, to get the URS it is necessary the continuous control philosophy and the procedural automation philosophy. The FRS as already defined builds its requirements and design according to URS and piping and instrumentation diagrams.

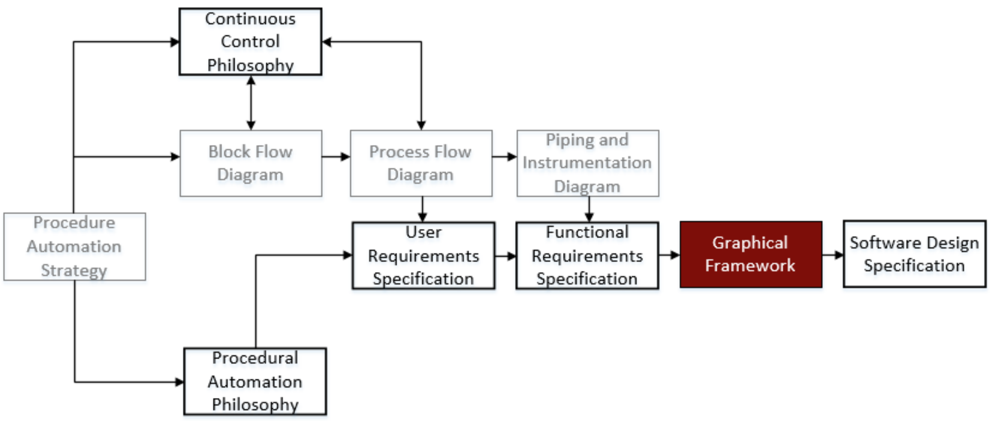


Figure 2.13: Graphical Modelling Framework Stage, own source.

2.5 Safety Instrumented Systems

The safety-instrumented systems have been part of the automatic protection systems since risks in process control systems has emerged. In recent decades, standards have been developed merging risk analysis with system design, using probabilistic failure analysis, and defining good engineering practice.[22]

A SIS is defined as a system composed of sensor elements, logic solver elements and final elements designed for the purpose of:

- Automatically taking an industrial process to a safe state when specified conditions are violated.
- Permitting a process to move forward in a safe manner when specified conditions allow (permissive functions).
- Taking action to mitigate the consequences of an industrial hazard.

A safety-instrumented system is used to implement one or more safety instrumented function.

2.5.1 Safety Instrumented Function

A safety instrumented function is a group of equipment intended to reduce the risk due to a specific hazard (a safety loop). The purpose of a safety-instrumented function is to automatically take the process to a safe state when specified conditions are violated. Figure 2.14 shows a safety instrumented function (SIS) with a basic process control system (BPCS).

Some safety instrumented function examples are:

- On detecting high temperature, prevent column rupture by shutting off steam flow to the re-boiler.

- On detecting high pressure, prevent tank rupture by an opening valve to relief system.
- On detecting high level, open drain valve to direct excess liquid to waste sump to reduce environmental damage.

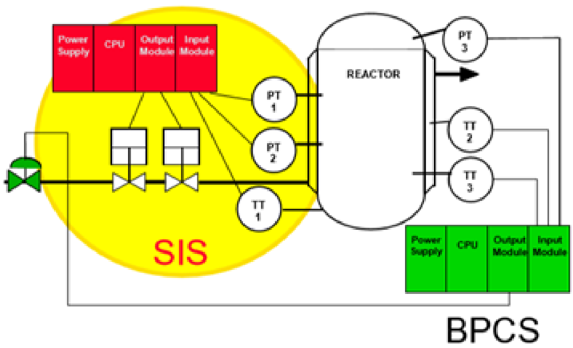


Figure 2.14: Safety Instrumented Function in BPCS, taken from [23].

2.5.2 Safety Integrity Level

The Safety- Integrity Level (SIL) is a discrete level classified by the requirements of the safety integrity functions to be allocated to the safety-instrumented systems [23]. The levels define how well the safety instrumented system performs its job of managing risk.

A safety instrumented system may have multiple safety instrumented functions with different individual safety integrity levels, so it is not correct to define a safety integrity level for an entire safety instrumented system.

If a process engineer needs to apply, safety instrumented system then should follow the standards powered by IEC. To apply safety in systems use the IEC 61511 standard [24].

2.5.3 Interlocks

An interlock prevents actions from taking place until pre-defined criteria have been satisfied or take actions to position the plant in a safe state.

For example, a tank outlet valve cannot open if the adjoining header is being cleaned In most applications, an interlock is used to help prevent a machine from harming its operator or damaging itself by preventing one element from changing state due to the state of another element, and vice versa [25].

The interlocks relate two mechanisms or functions mutually dependent.

This dependency could be between two mechanisms that block equipment or could be a dependency that blocks part of the process by blocking equipment or more than one equipment. So the interlocks are classifieds in:

- Process Interlocks: The process interlocks are characterized by stopping part of the process in order to avoid bad consequences.
- Equipment Interlocks: Represents an interlock that stops the operation of equipment.

It is important to highlight that the process interlock could be programmed to protect the process by stopping equipment but that does not mean that is an equipment interlock.

Chapter 3

Development Process

This chapter describes the process development of the graphical shapes and the way its components was defined.

This chapter also includes a description of the documentation that supports the GMF development and understanding.

The GMF has two main components, the control system component and the complexity component, each of it, are explained in this chapter subsections.

3.1 Control System Component Development Process

For the understanding and conceptualization of the GMF, it is important to remind that the scope of the framework is focused on identifying the complexity of the control systems associated with the continuous processes. In that way, it is decided to begin by exploring what a system represents in any field in order to assess what is the best way to represent the control systems and thus guarantee the compliance of the objective.

3.1.1 System

A system is a set of things working together as parts of a mechanism or an interconnecting network; a complex whole.

A system comprises of the following statements:

- Any system whatsoever is composed of a structure and a dynamic; the structure answers the question: What is the system? It means what components it has, and the dynamic answer the question: What does the system do? It means how the process is currently being executed by the system.

To the right performance of a system are necessary the following components:

- The logical programming that determines a sequential flow that the system should follow in order to fulfill a function.
- The system operation that is necessary for the system to fulfill the assigned function.

3.1.2 Structure

The following chart explains how the previous statements were organized in order to establish a path to structuring the GMF. This chart ties together the generic system components with what would be the components of a control system.

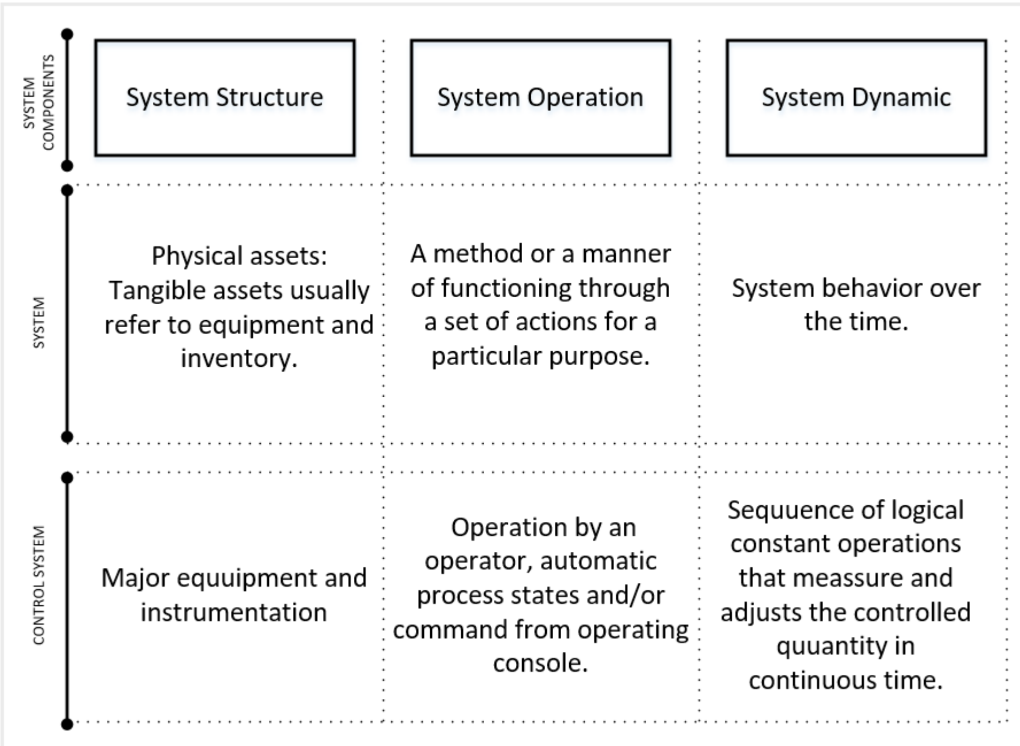


Figure 3.1: Path to structuring the Graphical Modelling Framework, own source.

The figure 3.1 relates three layers that expose:

- System components: As already described, a generic system component includes structure, operation and dynamic.
- Generic System: Describes what involves the structure, operation and dynamic in generic system.
- Control System: Describes what involves the structure, operation and dynamic in control system.

The following conclusions were drawn from figure 3.1:

- In order to have the control system structure associated with the continuous processes, it is necessary to identify the physical components that comprise the system through the ISA TR106.00.01 physical model.
- The system dynamic is given by the logical programming. In the case of control systems, logical programming is carried out by the continuous control strategy.

The control systems associated with continuous processes manage its system operation by an operator, process states and/or command from operating console.

After the previous conclusions, we proceed to explore each component to establish the structure of the GMF, focusing on gathering the necessary information to then be able to identify which components bring complexity to the processes. In that way the steps described below were taken:

- To review the physical Model and establish a conceptualization.
- To review the continuous control functionalities and components.
- To go deeply on operation interaction and states whit the control system associated to a continuous process.

All of these steps are based on ISA TR106.00.01 concepts and will be described in the following subsections.

3.1.3 Physical Model Review and Conceptualization

For the physical model review and conceptualization, it was necessary to explore and research continuous processes in order to apply the physical model and to find the best way to represent the physical assets in the GMF.

The figure 3.2, describes the physical model of tomato sauce process, this figure represents only the process equipment available for continuous production and does not include control algorithms that may be associated with them.

PLANT	PLANT AREA	UNIT	EQUIPMENT	DEVICE
Food Emulsions	Tomato Processing Plant	Unit Tomato Preparation	Infeed Conveyor	Weight Indicator
				Motor 001
				Gear Box
			Tomato Washer	Motor 002
				Motor 003
				Solenoid Valve 001
				Sorting Table
		Unit Juice Extraction	Tomato Juice Extractor	Hand Valve 002
				Variable Frequency Drive 001
				Motor 004
				Level Transmitter 001
				Agitator
				Outlet Valve
			Pulp Preheater	Transfer Pump 001
				Heat Exchanger
				Temperature Transmitter 001
				Motor Pump 002
			Pulp Storage cum Balance Tank	Charge Solenoid Valve 003
				Discharge Solenoid Valve 004
				Level Transmitter 002
		Tomato Paste Unit	Condenser	Balance Tank
				Graduated Flow Control Valve
				Feed Pump
				Condenser
				Concentrate Recirculation Pump
				Concentrate Discharge Pump
				Condensate Pump
		Vacuum Evaporation	Evaporator	Vapour Separator
				Vacuum Pump
				Vapour Duct
				Seal Water Tank
				Steam Supply System
				Service Water Line
				Air Supply Valve

Figure 3.2: Tomato Processing Plant Physical Model, taken from [27].

It was taken an example of food emulsion process because it represents the typical behavior of a continuous process.

This process usually comprises of reception section where the raw material is received and prepared by a washed section to remove outer dirt and foreign matter; then is transported to the processing and extraction section where the raw material is exposed to changes by some equipment. Some food emulsion uses another section such as evaporation section where a steam system is used to release some materials components.[26]

In figure 3.2, it can be noticed that the unit level comprises the operation focus for the main pieces of the process equipment, according to ISA TR106.00.01 the unit layer represents the process hardware that carries out one or more major processing activities.

The book: Process and Control Optimization, [11], points out: "We all know that it is time to stop controlling flows, pressures, and temperatures and to start controlling and optimizing pumping stations, boilers, and chemical reactors. In the next decade, hopefully, we will see the development of the universal software package for the various unit operations that can be adapted to specific plants just by the entering of design data for the particular process." This paragraph allows understanding that to guarantee functionality it is necessary to focus on process units.

In that way, it is concluded that **the GMF formation will be totally based on the unit layer of the ISA TR106.00.01 physical model.**

3.1.4 Continuous Control Functionalities Review

As already defined in subsection 2.3, continuous control is fundamentally based on the control loops that manipulate and manage the control variables.

In the process to review the continuous control functionalities, it is concluded that the continuous control is fundamentally based on control loops, because are the ones that measure, controls and acts upon the process variables. In that way, it became necessary to recognize what are the key information about control loops that offer enough information about the control system and that support the identification of the control system complexity.

It is possible to display the number of control loops that a unit have, but although that data can offer to the GMF reader an outlook of the continuous control applied in the unit, it is not enough, it is important to know how and why the control is applied in the unit.

As described in subsection 2.3, it exists loop types that emerge due to the process need, it means that identified a loop type offer information about how the control system is being applied in the process.

The loops have different control capabilities according to the process need. For example, a control loop regardless of its type can be useful to control the quality of a product or may be appropriate for the process control optimization. So in order to know the "why" or the purpose of each unit control loops it is defined a classification of four generic capabilities of a control loop.

- Process Control Optimization.
- Quality Control.
- Throughput Control.
- Reliability Control.

In addition to the control loops information, it is necessary to expose the variables that are being controlled by the control loops, as described in subsection 2.3.2. The process variables are the physical dynamic that involves the process and are the values that the control loops aim to control, so it is decided to show as a source of information the variable per control loop and to display the critical variables.

The critical variables are the support for the GMF reader to have a brief overview of the potential variables that could affect the production process largely.

By having the information of the control loops, its purpose and the variables controlled per loop, the reader can have an overview of how the process and critical variables are being monitored and how the process is being controlled.

3.1.5 Operator and Process States Interaction

As defined in subsection 2.4.7, the procedural control highlight in the fact that procedures are the set of instructions for each operation and that can be managed by an operator, or by a system. For example, a distillation tower may have one person managing the reboiler and a second person the reflux drum. These has an operator and used a set of procedures for each operation. These procedures would instruct the operators as to which valve to open or close and when/how this was to be performed, recently the procedure no longer had to describe if the valve needed to be opened or closed; rather, it only had to instruct when and what set points needed to be set.

ISA TR106.00.01 offer the path to establish states and transitions by applying procedural control good practices, this control type is exclusive of continuous control.

Procedural control establishes that the procedure only needs to request that the pump be started and all the necessary basic control be executed in the equipment implementation module [15]. As a result, the valves will open in the correct order and the procedure will be much simpler.

In that way, the process of conceptualization made for procedure automation tools was to discover a way to display the necessary information about the procedural control applied in the system and to identify how automated the procedures are, and how the control systems associated to the process works in procedural control.

The items in procedure conceptualization applied in the GMF include:

- Relating the steps and states of a unit procedure with the operator interaction to the procedure of the unit.
- Getting a key indicator that shows how automated is the unit procedure.
- Defining the automation styles that explain how automated is the unit procedure.

The next section shows the process in the GMF complexity component development.

3.2 Complexity Component Development Process

In the process to develop the component of complexity in the GMF, it was necessary to relate the structure defined in figure 3.1, particularly the relation of the system operation with the logical programming.

In that way, it is explored how to identify complexity in logical programming and complexity in system operation.

The process of establishing complexity was a focus on offering key indicators that in just one number could expose how complex is the control system associated with the continuous processes.

3.2.1 Complexity in Logical Programming

It is important to remind that logic programming is absolutely referring to programming in control systems associated with continuous process.

The logical programming in control systems is referring to the management and manipulation of the process variables through a programmable logic controller (PLC).

The PLCs are able to control the process through inputs and outputs, the inputs are the measured value, the command or other values that need to be manipulated in order to manage the process, the outputs are the desired values that modify the process by a valve, motor or other devices.

In that way the number of inputs and outputs (I/O) in a process represents the effort that needs to have the programmer,**the less complexity in logical programming, the less effort needed.**

The effort is the first concept related to logical programming complexity.

ISA TR106.00.02 introduce the reusability concept, which is also related to the number of inputs and outputs therefore with the effort concept. Reusability is defined as the use of existing assets in some form. In the logical programming of control system context, the assets are the I/O.

Reusability is the second concept related to logical programming complexity. The following subsections describe how effort and reusability concept where applied in the GMF, and how do they pretend to be useful to identify the control system complexity associated with the logical programming.

3.2.2 Reusability and Effort

The reusability and effort will be measured per units taking into account that the GMF formation is totally based on the unit layer of the ISA TR106.00.01 physical model.

Table 3.1 exposes a spreadsheet that shows the items that were taken to get a number that represents reusability and effort per unit.

Unit Tag	Unit Type	AO	DO	AI	DI	Total I/O	EF	EF*I/O	Reusability
Unit-T101	CM	5	10	20	20	55	1.3	71.5	0
Unit-T102	CM+	5	10	20	20	55	0.3	16.5	55
Unit-T304	CM+	5	10	20	20	55	0.3	16.5	55
Unit-T105	D	4	10	20	20	54	1	54	0
Unit-R201	CM	3	8	15	16	42	1.3	54.6	0
Unit-R203	CM+	3	8	10	16	37	0.3	11.1	37
Unit-C304	D	4	8	10	16	38	1	0	38
						336		224.2	185

Table 3.1: Reusability and Effort Spreadsheet

Effort Percentage	66.7(%)
Unit Reusability Percentage	55(%)

Table 3.2: Reusability and Effort Percentage

For understanding the process to get reusability and effort number, it was taken a generic example.

The first two columns, unit tag and unit type, are intended to recognize and classify the unit. The columns in between the unit type and total I/O are the analog outputs (AO), digital outputs (DO), analog inputs (AI), digital inputs (DI), and the total of I/O that involves each unit with its control system.

The unit type column corresponds to a classification based on the concept of modularization made by ISA TR106.00.01. To see this concept, please refer to subsection 2.4.4.

ISA TR106.00.01 introduces what control implementation module and equipment implementation module represents in the environment of logical programming for the control system, in that way the table 3.3 exposes the classification made, based on modularization proposed by ISA TR106.00.01. The classification is directly related to an effort factor defined according to modularization concept.

Unit/Module Objective	Unit/Module Type	Type Tag	Effort Factor
Created to Reuse	Control Implementation Module	CM	1.3
	Equipment Implementation Module	EM	1.3
Applied from Reuse	Applied Equipment Module	EM+	0.3
	Applied Control Module	CM+	0.3
Stand Alone	Unit or Module	D	1

Table 3.3: Effort Definition, own source.

This classification is based on the assertions of the logical programming that involves the control system associated with the unit or the implementation module. It means a unit or module logical programming could be created in order to be reusable by make the program available for being applied in other automation projects that involves similar features. A usual strategy to have a unit or module available to be reused is to make it parametrizable which represents a powerful technique for the reliable reuse of software. In this technique, modules are parameterized over very general interfaces that describe what properties of an environment are required for the module to work correctly.

To create an implementation module or unit to be reused increase the effort that needs to be applied by the programmer.

On the other hand, the fact to use a unit or module already created to be reused, decrease the effort that needs to be applied by the programmer. It also exists module or units that are just programming without any intention to be reused. As a conclusion, it was defined an indicator called effort factor, that represents the necessary effort to create a unit or module according to its objective. The indicator shows:

- 1 or 100 % is the effort needed to create a module or unit created **without any intention to be reused.**
- 1.3 or 130 % is the necessary effort to create a module **available to be reused.**
- 0.3 or 30% is the effort needed to **apply a module or unit already created to be reused.**

The fact of being varied by the 30% will depend upon the integrator or the group of people that model the plant, this value could depend on time in programming, cost, people needed and other variables. The definition of this value is out of the scope of the GMF, therefore, it was taken the 30% as a percentage example.

The column EF*I/O, represents a necessary equation to determine the effort needed to create the plant area. The following equation show how to get the effort in a quantitative way.

$$\frac{\sum EF * I/O}{\sum TotalI/O} = Effort\,Percentage \tag{3.1}$$

Finally, the last column called reusability, just point out the I/O classified as reusable, it means the ones that belong to the unit type of "applied from reuse", then the following equation is applied.

$$\frac{\sum Reusability}{\sum TotalI/O} = Reusability\, Percentage \tag{3.2}$$

3.2.3 Complexity in System Operation

The system operation is referring to the way in which the system is being operated; it could be by an operator interaction or it could be by an automatic operation as explained in subsection 3.1.5.

An example of operator interaction in the continuous process could involve a selection of the correct pump to be used for specific major equipment, or a push button that represents a turn-on of a control panel or a final control element. On the other hand, an automatic operation could involve a status change by an automatic transition to the next step when all the above equipment is running.

Hence, the complexity in operation is based on the number of operator interactions per steps with the system, it means as a first conclusion, **the more the system needs of human intervention the more complex in operation.**

In continuous processes could exist an abnormal situation in which the operator interactions increase. An abnormal situation could refer to an undesired state due to wrong behaviors of equipment or undesired values of variables when that situation occurs the system interlock the necessary equipment or group of equipment to avoid bad consequences. In the instance where an interlock is triggered, human intervention is necessary to return the system to the normal operating state.

In that way as a second conclusion, **the fact of having at least one interlock increase the probability that complexity in system operation increase.**

For more information about interlocks, please refer to subsection 2.5.3.

For the GMF it was relating the operation interactions and the automatic operation (system operation) information with the interlocks data.

This component shows the number of process interlocks and equipment interlocks present in the unit. An interlock is a mechanism that prevents or stops equipment actions when a predefined condition has been met to keep the equipment or the process in a safe state. The interlocks can be classified in:

- Process Interlocks: Stop the process or part of the process (or put the process in a safe state) when specific conditions are met in order to avoid undesired consequences.

- Equipment Interlocks: Stop the operation of specific equipment (or put the equipment in a safe state) when specific conditions are met in order to avoid undesired consequences.

The figure [3.3], shows the interlock representation in the diagram.

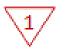
Symbol	Symbol Name	Symbol Description
	Interlock Generation	The unit generates an interlock that affects other units. The number shown is a unique interlock identifier in the diagram. This number allows to identify the units affected by the interlock (by the Interlock Impact symbol).
	Interlock Impact	The unit is impacted by an interlock generated in another unit. The number shown is the unique interlock identifier that allows to identify the unit that generates interlock (link with the Interlock Generation symbol).

Figure 3.3: Interlock representation,own source.

In the final instance, due to the identification of the system interactions is done from the steps it was relating two concepts already defined in subsection 2.4. The relation of these concepts is defined in the following subsection.

3.2.4 State-Loop Based Control and Automation Style

In order to identify the operator interaction and the automatic operations involved in a procedure, it is necessary to understand the procedure components.

According to ISA TR106.00.01, Procedures can be associated with either mode of operations or process states. Procedures can be designed to perform process transitions, such as start up a unit, transition a unit to a new production rate, or perform a controlled shutdown on a unit.

The figure 3.4 shows a procedure scheme that represents an example of a procedure that conceptualizes the following terms definition that composes a procedure according to ISA TR106.00.01.

- Process State: A state is defined as a definable operating condition of process equipment as it progresses from shutdown to operating and back to shut down. Each process state represents a unique operating regime that supports the process equipment's objectives of processing an input into the desired output. When using process states, procedure automation is centered on a major piece of process equipment, usually a unit.

  Example: A unit's maintenance wait (down for maintenance), a dual pump set's process wait (getting ready for startup or maintenance) and a plant area's start-up are examples of process states.

- Step: The steps are the description of the actions that the system or the operator should follow to go to one state to another, it could be related with the step by step instructions to accomplish the necessary behavior to perform the procedure. Example: The actions of a turn on the equipment, close valves, disable analog alarms, starts a flow control are examples of steps. These steps could be performed by the system or by the operator.

- Transition: The transitions are the conditions to go from one state to another, the conditions should be a statement that describes the necessary parameters to enable the system to the other state. Example: A transition could be:

System transition to the next state when the outlet temperature is greater than x F. System transition to next state when combustion air is within +-x% for x seconds Operator enables the transition to next state when all the above equipment is started.

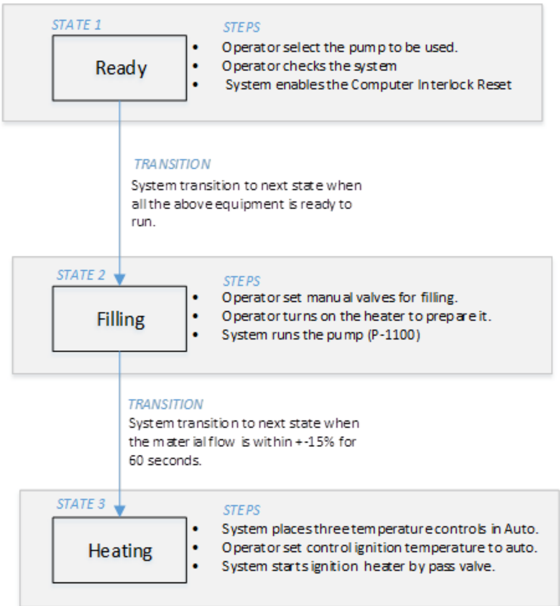


Figure 3.4: State Transition Diagram, own source

The state transition diagram can be a useful tool to help decide the most appropriate automation style to apply. The actions that are to be performed and verified to transition the process between steady-state conditions can be associated with each transitory state. This enables engineers and operations personnel to visualize the work required and to help decide whether the actions are to be done by an operator or by the system.

The use of process states is one method that gives a framework for organizing automated procedures as more complex automation styles are used. This type of procedure automation is called state-based control.

Thereby the GMF includes the automation style and state based control component in system operation complexity.

1. Automation Style: The automation style component shows how automated is the unit and the relation between the system, the process, and the operator. The following are the automation styles were defined:

   (a) MO – Manual Operation: The operator executes all the tasks. There is no interaction with the control system.

   (b) AMO – Assisted Manual Operation: The control system suggests alternatives of the tasks to perform and the operator decides and executes the tasks based on the suggestions.

   (c) SO – Shared Operation: The control system suggests the tasks to perform and, if the human operator approves, the system executes the tasks.

   (d) AO – Automated Operation: The control system automatically executes the tasks and follows established procedures. The operator interaction is limited to system failures or plant stoppages.

   (e) FAO – Fully Automated Operation: The control system automatically executes all the tasks and follows established procedures, including disturbances fixing, without any operator interaction.

2. State Based Control Component.

   (a) The unit implements procedural control based on states (e.g. waiting, mixing, filling, etc.). This strategy is used to organize automated procedures that allows transitions between the states when a set of conditions are met. The system executes commands by itself based on the process states.

   (b) The unit applies loop-based control. In this strategy, the operator has the responsibility to execute commands to keep the variables in the desired values.

Chapter 4

Graphical Modelling Framework Tools

The purpose of creating a support tool and documentation for the GMF is to give a detailed description of the use of the GMF and offer the necessary information to allow understanding of the GMF concepts.

This chapter explains the process of documentation development and clarifies what makes up each document or support tool. The support tools and documentation are not a deliverable for the University of Cauca, because are protected for a confidentiality agreement; hence, this documentation will be briefly shown in the oral presentation.

The GMF support tools and documentation are the following:

4.1 Guide to document automation functionalities by using the GMF based on Omnicon client defined approach.

The guide to document automation functionalities by using the GMF is a document whose components were developed according to Omnicon client defined approach, the guide includes:

- **Research Report:** It is a report that aims to provide information concerning the main concepts acquire after the research on technical reports and articles related to procedure automation for a continuous process. The document provides relevant information and leading practice of procedure automation for the continuous process from ISA 106 technical report.
- **GMF *Playbook*:** This document introduces, describes and display the GMF by offering the concepts explanation, diagram notation, diagram symbols, and the technical concepts definition applied in the diagram.
- ***"How-To"* Guide:** This document provides key information in order to define control system for unit operation, this guide sketch outline the key factor required to establish a comprehensive process control strategy based on ISA TR106.00.01 for continuous processes.

4.2 Graphical Shapes

The graphical part of the GMF is called shape.

The concept of shape in graphical modelling context comes from Microsoft Visio Drawing Tool that calls "shapes" to the figures that represent different types of actions, steps in a process, or shows information about a specific topic. Visio shapes are ready-made objects that are dragged onto the drawing page.

Thereby it was developed graphical shapes intended to show clear and specific information about the control systems associated with continuous processes that achieve a general comprehension to allow the engineers or operators to identify its complexity.

The shapes development process was focus on show the information in a way that the complexity of the control system could be recognized in a quick overview and to offer enough information to relate the data with key indicators that allow the GMF owner to understand the type of automation functionalities of the entire system. It used the Microsoft Visio Drawing to develop the following toolkits, with the purpose of making a standard for the GMF user's manipulation:

- **GMF Visio Library:** A Visio library is a set of pre-developed master shapes and is used to achieve more functionality. The Visio library can just be called within the program body without defining them explicitly. The Visio library is already saved in *Visio Stencils* [27] in order to be available to graphical modelling development.

  Note: When a shape is dragging from the Shapes window onto the drawing page, the original shape remains on the stencil. That original shape is called a master shape. The shape that is putting on the drawing is a copy. It can be dragged as many instances of the same shape onto the drawing as necessary.These shapes were developed with special features of Visio shapes.[28]

- **GMF Visio Templates:** A Visio template is a layout setting that has unique features that allow representing the graphical shapes in a standard display that has been already defined.

- **GMF Job Aid:**The job aid is a document that explains the steps to use the graphical modelling items to create the GMF by using Visio tools such as templates, stencils, and library. The job aid is designed to reduce avoidable mistakes by establishing what to do.

4.3 GMF Example

The GMF was applied in an example in order to have a tool to validate the concepts and uses of the GMF.

The process to apply the GMF was first to find in Omnicon database of projects, a real plant that runs continuous processes and that comply with the following parameters:

- Comprehensive and in-depth information on control systems.
- Approximate control software development based on states.
- Defined as complex based on the criteria of the project developers.

It was selected a plant that produces chemical solutions located abroad.

After defining the plant that was taken as an example to apply the GMF, the following steps were established:

- Gather the available documentation such as:

  Design Diagram: PID, PFD, Control Loops diagrams, Equipment Layout, Floor plan, SAMA.

  Control Documentation: HMI scheme, PLC backup, Control Philosophy, Operation Manual, Control Narratives, SOPs, Trainning Material.

  Process Safety Management Documentation: Process Hazard Analysis, Hazard and Operability Studies, Quantitative Risk Assessment.

- Explore documentation by following the established spreadsheet tool that relates each document with a specific checklist that helps to visualize which information ought to have each documentation and which information will be useful for the GMF application.
- Follow the "*How to* " Guide to structure the information already gathering.

- Use the GMF *Playbook* to prepare the GMF application.
- Follow the job aid to establish the graphical shapes and develop the GMF diagram.
- Make conclusions about the complexity of the control systems associated with the chemical plant.

4.3.1 Example development

After the information gathered and the steps followed, it was defined the continuous processes in phases description.

Each phase meet a common goal to produce a chemical substance that is needed to get a neutralized product, as a first stage the main substance to produce is So3, so firstly the process produce So2, to produce this, the cooler and sulfur get ready to sulfur recirculation, combustion and atomizing air and send it to the burner that in parallel to this process has been pre heat by the heater to ignite the temperature, the sulfur flow ignites the flame and the flow is controlled to reach the temperature set point, the SO3 converter receives the sulfur from the burner, which submits the substance at a variable temperature, finally the SO3 produced goes to the oleum demister to enhance the removal of liquid droplets entrained in a vapor stream.

Once the S03 is already on line, the organic feed enter to the reactor and the SO3 path changes sending SO3 to the sulfonator that generates sulfonic acid, the raw material for neutralized products, and to the acid absorber that is responsible to catch the acid that comes from S03 production. Subsequently the sulfonic acid is mixed with caustic, water, blender, alcohol and additive in order to homogenize and create a neutralized product, followed this the product result is mixed in a second pump and sent to the deaerator (the deaerator is optional and depends of type of product).

It is important to highlight that before the process starts, the caustic scrubber needs to be ready, and the acid absorber phase could be started before starts SO3 production.

According to the phases described it was defined the following unit modules:

1. Product A Unit
2. Converter Unit
3. Reactor Unit
4. Absorber Unit
5. Neutrilizer Unit
6. Storage Unit
7. Release Unit

In the Figure 4.1 it can be notice the equipment that involve each unit module, and the overview in high level of the process flow.

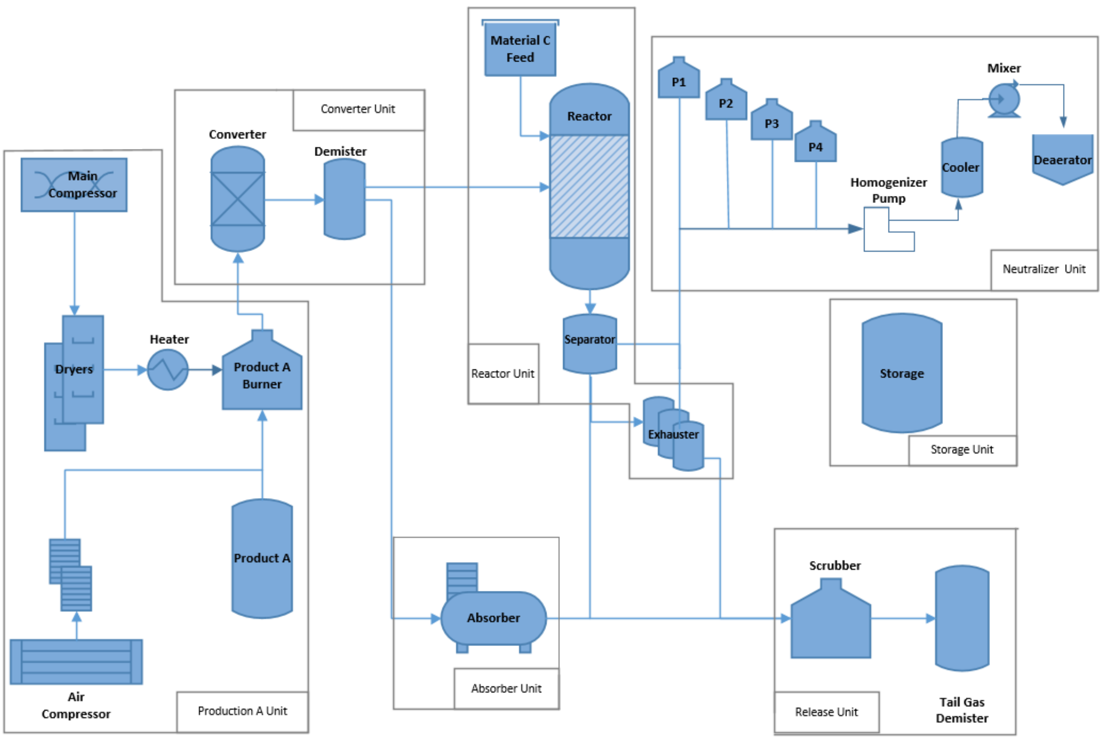


Figure 4.1: Unit Modules, own source.

In order to identify the elements needed to build the graphical component, it was necessary to establish the behaviour of each control system, as shown in the following example where it was defined the ignition air system that is implemented between the dryers, heater and product A burner where product A refers to sulfur:

Ignition System: The Ignition Air system consists of Process Air coming from the existing process air dryers. During the Sulfur ignition Combustion Air is controlled by PID loop. FIC 20001 opens the Ignition Air Valve FCV 20001 based on the set point and the Combustion Air Flow Transmitter FT 20001 feedback. The system will have a Combustion/Ignition Air rupture disc PSE 20011 specified to indicate high pressure in the Combustion/Ignition Air line.

The temperature of the air from the Ignition Air heater VE30701/SP30701 will be controlled by JC30701 (Heater SCR Power Level) based on the set point and feedback from the Ignition Air Heater Outlet Temperature Transmitter TT30701. The Ignition Air Heater control is activated/deactivated by JY30701 (Ignition Air Heater ON/OFF). The Ignition air heater will have a heater element Temperature Transmitter TT30702 with a high temperature alarm and interlock to shut down the Ignition Air Heater, if the temperature reaches >1250F. There is an ignition air heater temperature transmitter at the midpoint of the heater, TT30703, to monitor the external surface temperature of the heater.

The sulfur surner is isolated from the process air system by the combustion air block valve XV30703. An ignition air heater by-pass valve XV20002 is interlocked such that if the ignition air heater is ON it is closed and when the ignition air heater is OFF it is open. By-passing air around the heater is necessary to avoid loss of combustion air pressure due to high flow of combustion air through the heater during normal sulfur burning rates.

In the Figure 4.2, it can be notice the elements described.

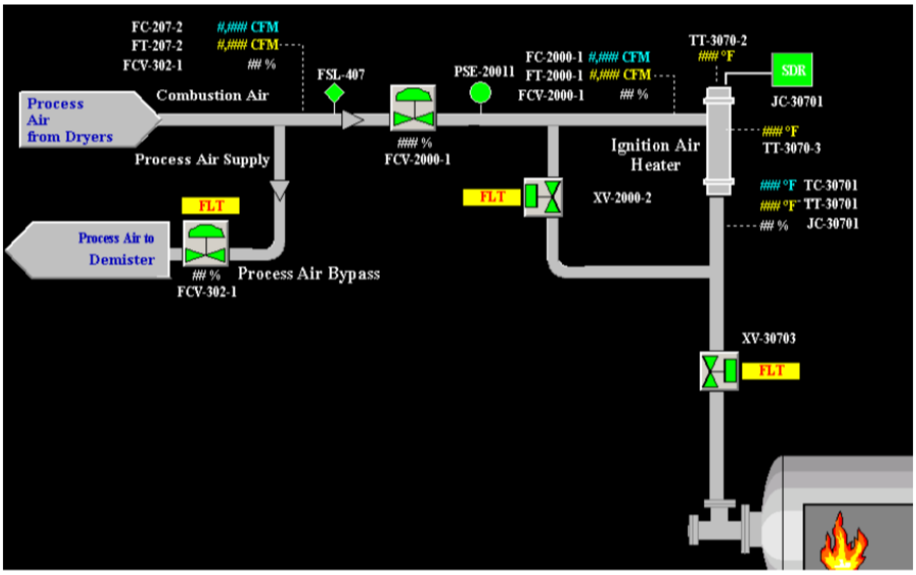


Figure 4.2: Ignition System, taken from process HMI.

Subsequently it is recognized the equipment and process interlocks by using the PIDs, some of the information gathering in control and process narratives and with the experience of the control support engineers of the process.

The Figure 4.3 shows part of a PID that shows the interlocks and table 4.1 shows how the interlocks where organized in order to establish its dependency between unit modules.

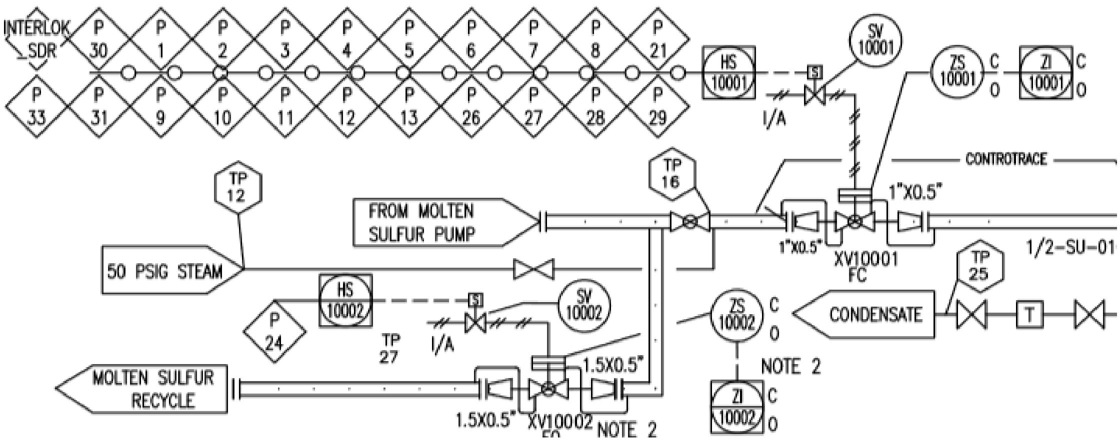


Figure 4.3: Interlocks in a P&ID, taken from process diagrams.

# Interlock	Equipment or Process Conditions that initiates Interlock	Actions	Logic Required	Unit(s) Involved
P01	Burner high temperature fault	Gas plant shutdown sulfur and air flow stopped. Absorber and Scrubber remain running.	Stop P-102-1 &-2; Stop B202-1;	Production A Unit

Table 4.1: Interlock table information,own source.

After the unit modules,loops and interlocks identification, it is recognized the number of inputs and outputs (I/O) analog and discrete for each unit module, by using the PIDs and the control narratives. The table 4.2. shows part of the (I/O) recognized.

Tag Name	Devices	Description	DI	DO	AI	AO
CA 10201	Internal Calculation	Sulfur / Air Mass Ratio			1	
FT20001	Transmitter	Combustion Air Flow			1	
TT30702	Transmitter	Ignition Air Heater Inlet Temperature			1	
TT30701	Transmitter	Ignition Air Heater Outlet Temperature			1	
TT30703	Transmitter	Ignition Air Heater Element Temperature			1	
FT30101	Transmitter	Atomizing Air Flow Transmitter			1	
ZILL20011	PSE20001 BURST	Combustion/Ignition Air Rupture Disc	1			

Tag Name	Devices	Description	DI	DO	AI	AO
ZILL30711	PSE30701 BURST	Nitrogen Air Rupture Disc	1			
ZIC30703	Closed limit Switch	Combustion Air Block Valve	1			
HS30704	Valve	Nitrogen purge Valve		1		
JY30701	Air Heater ON/OFF	Ignition Air Heater ON/OFF		1		
JC30701	Analog Valve	Heater SCR Power Level				1

Table 4.2: Inputs and Outputs Recognition,own source.

Finally to define the components that are part of the complexity in operation it was necessary to follow the phases defined and establish states per unit model as showing bellow:

The SO3 Gas Plant production phase currently is programmed with 8 States:

- State 1 System Checks.
- State 2 SO3 Cooling Started.
- State 3 Sulfur Recirculation.
- State 4 Pause prior to loading compressor.
- State 5 Load Compressor.
- State 6 Start burning Sulfur.
- State 7 Continuing bring system up to temperature.
- State 8 All equipment halted.

State 1	System Checks
Step 1	Permissive to Continue. Operator prompts While pausing: Operator Acknowledgement RequiredChecking Caustic Scrubber & Acid Absorber Is the Caustic Scrubber Ready? (State 4) Is the Acid Absorber Ready? (State 3)If resumed. but not advancing: Still Waiting for Scrubber & Absorber
Step 2	Operator Prompts While pausing: Operator Acknowledgement Required Set Manual Valves for SO3 Cooler Set Manual Valves for Sulfur Recirculation & Air Select Sulfur Pump & Turn on Dryer Panel Operator selects the sulfur pump to be used
State 2	SO3 Cooling Started
Step 1	Operator Prompts Starting S03 Cooling
Step 2	PCS places these three temperature controls in AUTO: SO2/AIR Cooler SO2 Vapor temperature control (TCV-301-2) at a set point of 820 F SO3 converter 2nd pass temperature control (TCV-301-5) at a set point of 850 F S03 converter 3rd pass temperature control (TCV-30I -7) at a set point of 850 F 3. PCS runs the SO3 cooling water pump (P-302-1)
Step 3	PCS starts the SO3 cooling / H2O temperature control (TCV-302-1) at a set point of 107 F PCS starts Air Dryers (DRYER_EN) in Automatic control PCS

Table 4.3: Steps and States Identification, own source.

Finally according to the information gathered it is applied the equations defined in subsection 3.2.2, the tables 4.4, 4.5 and 4.6, shows the respective data to get reusability, opperation and effort percentage.

Reusability Data

UNIT NAME	INPUTS AND OUTPUTS	REUSABLE INPUTS AND OUTPUTS	% REUSABILITY
Product A Unit	37	5	14%
Converter Unit	12	11	98%
Reactor Unit	14	9	61%
Absorber Unit	4	2	52%
Neutrilizer Unit	7	4	64%
Storage Unit	3	2	98%
Release Unit	5	2	45%

Table 4.4: Data to get Reusability Percentage,own source.

Opperation Data

UNIT NAME	OPERATOR INTERACTIONS	# STEPS	% OPERATION
Product A Unit	16	56	52%
Converter Unit	11	14	82%
Reactor Unit	13	23	58%
Absorber Unit	3	4	82%
Neutrilizer Unit	16	29	54%
Storage Unit	0	1	5%
Release Unit	1	6	18%

Table 4.5: Data to get Opperation Percentage, own source.

Effort Data

UNIT NAME	I/O	Eff*I/O	% Effort
Product A Unit	37	34,78	94%
Converter Unit	12	1,44	12%
Reactor Unit	14	8,68	62%
Absorber Unit	4	3,56	89%
Neutrilizer Unit	7	3,36	48%
Storage Unit	3	0,42	14%
Release Unit	5	3,65	73%

Table 4.6: Data to get Effort Percentage, own source.

The diagram shows in Figure 4.4 , establish the overview of the complexity of the control system associated to the continuous processes, and the diagram shows in Figure 4.5.

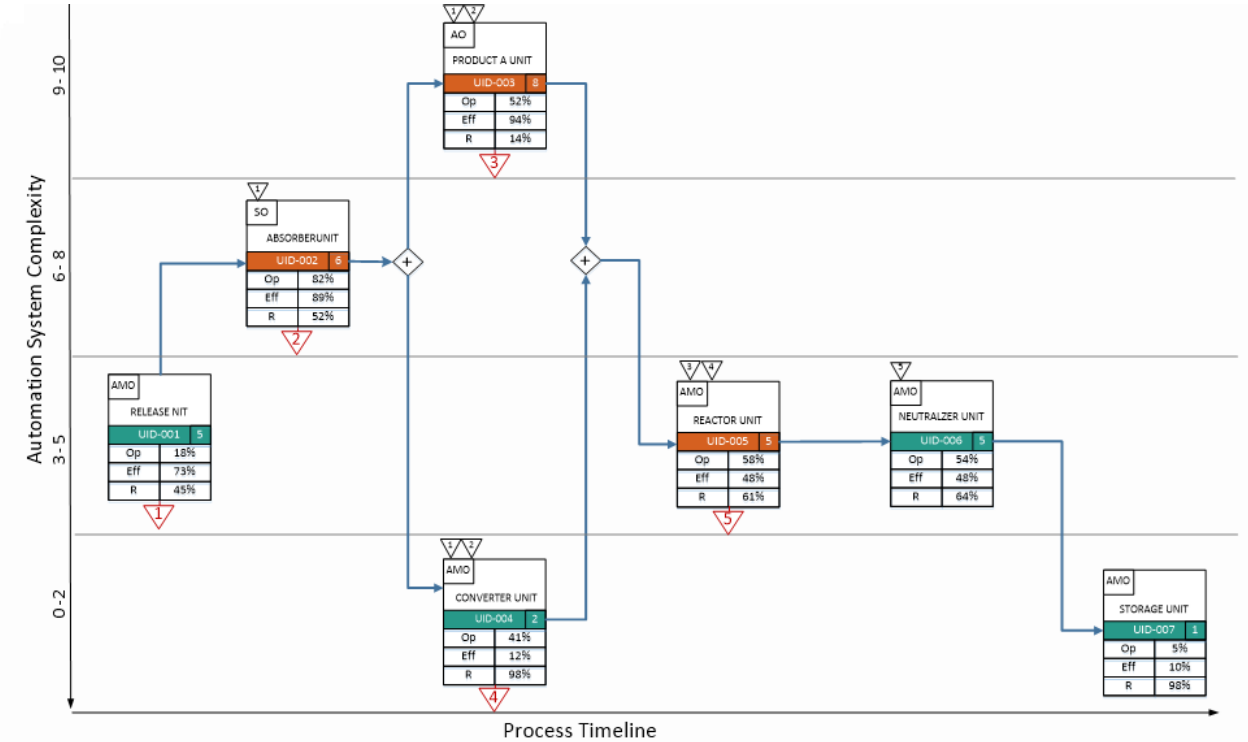


Figure 4.4: Complexity Overview, own source.

Chapter 5

Conclusions and Future Work

In this chapter, it is provided the main conclusions obtained through the evaluation. Finally, it is outlined directions for future work.

5.1 Conclusions

- The GMF allows process control and automation engineers to get an overview of the high-level control system capabilities without reading code or all the process narratives or control narratives.

- The GMF allows process control and automation engineers to explain, discuss and agree with manufacturing technology engineers and local plant operations staff about the designed control system.

- The fact to know the complexity of the control systems associated with continuous processes in a design stage offers the intended audience to support an accurate quotation.

- There may be several definitions of complexity; however, when discussing systems with components that can be evaluated objectively, it is necessary to establish parameters based on quantitative measurements that support a standard definition of complexity in control systems associated with continuous processes.

- It must be taken into account that the supporting documentation must be followed promptly in order to guarantee the desired results.
- By following the GMF documentation in a design stage will be ensure that the system and operator interactions are fit for purpose for the intended process and products.
- Based on the internship experience in creating deliverable for a client in a project that have no precedents it is necessary to have a constant validation with the client, in order to guarantee objective parameters that are representative for the utility of the component.

5.2 Future Work

- To develop an Automation Mapping Methodology Framework to document the current state of the control systems of a site in order to offer a diagnostic related to control system, with the purpose of applying the good practices defined in ISA TR106.00.01.
- To validate the GMF on site mapping in order to guarantee its functionality in a standard way.
- To develop documentation for establishing how to program vendor neutral including ISA TR106.00.01 coordination layer explained for Rockwell Automation and Siemens.

Bibliography

[1] M. Scheidgen, "Textual modelling embedded into graphical modelling," in *European Conference on Model Driven Architecture-Foundations and Applications*. Springer, 2008, pp. 153–168.

[2] B. Silver, *BPMN Method and Style, with BPMN Implementer's Guide: A structured approach for business process modeling and implementation using BPMN 2.0.* Cody-Cassidy Press Aptos, 2011.

[3] M. Fowler, C. Kobryn, and K. Scott, *UML distilled: a brief guide to the standard object modeling language.* Addison-Wesley Professional, 2004.

[4] T. V. Huynh and J. S. Osmundson, "A systems engineering methodology for analyzing systems of systems using the systems modeling language (sysml)," *Department of Systems Engineering, Naval Postgraduate School, Monterey*, 2006.

[5] Eclipse. (2019) Graphical modeling project. [Online]. Available: https://www.eclipse.org/modeling/gmp/

[6] BPMN. (2010) Quick guide second edition. [Online]. Available: https://www.bpmnquickguide.com/view-bpmn-quick-guide/

[7] S. A. White, "Process modeling notations and workflow patterns," *Workflow handbook*, vol. 2004, pp. 265–294, 2004.

[8] S. D. Schaber, D. I. Gerogiorgis, R. Ramachandran, J. M. Evans, P. I. Barton, and B. L. Trout, "Economic analysis of integrated continuous and batch pharmaceutical manufacturing: a case study," *Industrial & Engineering Chemistry Research*, vol. 50, no. 17, pp. 10 083–10 092, 2011.

[9] G. Tapperson and T. A. Boyd, "Distributed control system having central control providing operating power to wireless transceiver connected to industrial process control field device which providing redundant wireless access," Oct. 28 1997, uS Patent 5,682,476.

[10] Z. Zhou, S. Cheng, and B. Hua, "Supply chain optimization of continuous process industries with sustainability considerations," *Computers & Chemical Engineering*, vol. 24, no. 2-7, pp. 1151–1158, 2000.

[11] B. G. Liptak, *Instrument Engineers' Handbook, Volume Two: Process Control and Optimization.* CRC press, 2018.

[12] C. A. Smith and A. B. Corripio, *Principles and practice of automatic process control.* Wiley New York, 1985, vol. 2.

[13] J. G. Á. Díaz, J. Moreno, and F. Ramírez, "Diseño e implementación de un sistema de control cascada en la planta de intercambio térmico-pit000," *Informador técnico*, vol. 81, no. 1, pp. 32–43, 2017.

[14] A. C. Solé, *Instrumentación industrial.* Marcombo, 2012.

[15] C. ISA106, "Isa-tr106.00.01-2013 procedure automation for continuous process operations - models and terminology," International Society of Automation, Tech. Rep., 2013.

[16] W. Hawkins, D. Brandl, and W. Boyes, *Applying ISA-88 in discrete and continuous manufacturing.* Momentum Press, 2010, vol. 2.

[17] D. A. Huffman, "Benefits of state based control," *Isa. org, White Paper*, 2009.

[18] H. Bloch, S. Hensel, M. Hoernicke, K. Stark, A. Menschner, A. Fay, L. Urbas, T. Knohl, and J. Bernshausen, "State-based control of process services within modular process plants," *Procedia CIRP*, vol. 72, pp. 1088–1093, 2018.

[19] C. ISA106, "Isa-tr106.00.02-2017 procedure automation for continuous process operations - work processes," International Society of Automation, Tech. Rep., 2017.

[20] G. ISPE, "5: A risk-based approach to compliant gxp computerized systems," in *ISPE*, 2008, p. 207.

[21] Y. Nazer, "A chemical and petroleum industries division (chempid) webinar," *As part of the annual CHEMPID membership campaign*, 2016.

[22] J. L. Bergstrom, "Safety instrumented systems (sis) and safety life cycle," *Process Engineering Associates, LLC*, 2009.

[23] B. Standard and B. IEC, "Functional safety—safety instrumented systems for the process industry sector—," *ANSI/ISA S*, vol. 84, 2003.

[24] S. Hauge, P. Hokstad, and T. Onshus, "The introduction of iec 61511 in norwegian offshore industry," in *Proceedings of the European Safety & Reliability International Conference (ESREL'01)*, 2001, pp. 483–490.

[25] . Roy E. Sanders, in Chemical Process Safety (Fourth Edition). (2019) The role of mechanical integrity in chemical process safety. [Online]. Available: https://www.sciencedirect.com/topics/engineering/interlock

[26] W. A. Gould, *Tomato production, processing and technology.* Elsevier, 2013.

[27] B. Biafore, *Visio 2007 Bible.* John Wiley & Sons, 2007, vol. 408.

[28] M. Office. (2019) A beginner's guide to visio. [Online]. Available: https://support.office.com/en-gb/article/a-beginner-s-guide-to-visio-bc1605de-d9f3-4c3a-970c-19876386047c

List of Abbreviations

BPCS	Basic Process Control System
BPMN	Business Process Model and Notation
CCM	Complex Control Implementation Module
CM	Control Implementation Module
CS	Cascade
Cx	Complexity
Eff	Effort
EM	Equipment Implementation Module
FB	Feed Back
FF	Feed Forward
GAMP	Good Automated Manufacturing Practice
GMF	Graphical Modelling Framework
ISA	International Society of Automation
LBC	Loop Based Control
OL	Open Loop
PID	Proportional–Integral–Derivative
PLC	Programmable logic controller
R	Reusability
STB	State Based Control

SIF	Safety Instrumented Function
SIL	Safety Integrity Level
SIS	Safety Instrumented System
TR	Technical Report
UML	Unified Modeling Language