OPEN APPLICATIONS GROUP INTEGRATION SCENARIOS

Introduction

This chapter depicts the sample integration scenarios that the content of the Open Applications Group Integration Specification (OAGIS) supports. These scenarios are intended to be used as models for designing ones own solutions based on the integration content in OAGIS. They are not meant to be used only as shown.

These models may be used as examples and modified for the specific needs of your organization.

Please Note:

Some of these integration scenarios have details concerning their workflow, assumptions, and exception requirements while others do not. We are working to add these details quickly. The information exists, it is in the process of construction.

If you have any questions concerning these or the use of them, please contact us at info@openapplications.org. Thank you very much for your patience in this matter.

1.0 GENERAL LEDGER TO SUB-LEDGER SCENARIO DESCRIPTION

1.0 Overview

This chapter describes the integration scenario for sub-ledger business software components to integrate with a general ledger business software component.

The purpose of this scenario is to enable the visualization of the components and the dialogs between components for this specific integration domain. This scenario is *not* meant to be the only correct model for integrating sub-ledger components to a General Ledger business software component. This is one model that may be used to design one's own integration based on the specific needs of an organization or group of organizations.

Many applications create data that cause changes in the account balances of a general ledger application. Some components that have activity which will be reflected in a general ledger application are:

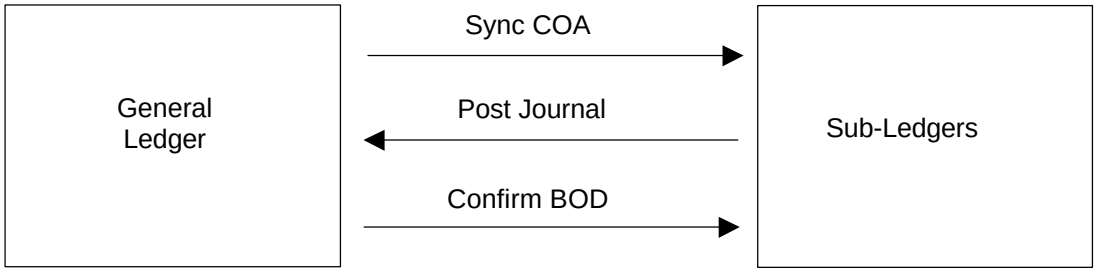
1. Benefits
2. Costing
3. Human Resources
4. Inventory Control
5. Manufacturing
6. Payroll
7. Production
8. Treasury

This is not a complete list but is meant to be a representative sample of activities that generate a journal entry.

1.1 Scenario Diagram

The scenario diagram below contains the components involved in the interaction, the dialog flows or conversation between the components, certain assumptions about the sequence of events, and assumptions about the technical approach, for example, publish and subscribe. The next three sections will give a brief overview of the components, the sequence of events, and the environment.

Again, this is a MODEL to be used as a reusable design document, not a required approach.



Accounts Payable
Accounts Receivable
Budget
Project Accounting
Manufacturing
Inventory
Order Management
Billing
Purchasing
Assets
Human Resources

1.2 Assumptions

This scenario assumes a loosely coupled, asynchronous approach to the integration. Transaction management is implicitly required but not explicitly defined.

This scenario describes a model for one or more sub ledger components integrating with a common general ledger component. The environment for this integration may be within a single enterprise, or across several divisions. There may be instances where the amount of information sent to the general ledger is enough, and they may be times when the sub ledgers need to send the detail transaction data to a cost accounting component.

This scenario does not cover posting to a cost accounting component. This scenario also assumes that the details of the financial transactions will be kept in the sub ledgers and the drill back mechanism from the general ledger component to the sub ledger components is described in the POST JOURNAL Business Object Document in chapter 1, section three of OAGIS.

1.3 Component Definitions

This scenario contains two major components, the general ledger component and the sub-ledger component. The definitions of these components are left to the designer but are assumed to contain the functionality as defined by what is commonly sold in the commercial market place today.

This heuristic definition is broadly accepted by the designers of the integration scenario and is a direct decision not to define how the processing takes place within any individual component. The component must be able perform the services defined by the business object document, but the internals of the component are not required or desired to be exposed.

The most important factors in defining these components is to ensure that the designer can communicate the requirements and design precisely enough to design the interfaces needed and their interrelationships.

1.4 Business Workflow (Sequence)

This scenario has two major events in the workflow sequence.

1) The first business event in the sequence may be the synchronization of the financial chart of accounts. This scenario assumes that the general ledger component “owns” the chart of accounts definition and the instances of data within it. The sub ledger applications have several choices for validation of account numbers and other fields in the complete chart of accounts structure. One of these choices is to synchronize the chart of accounts structure and data from the general ledger application to all of the sub ledgers.

The synchronization process can ensure that all of the components in the integration scenario will have the same data necessary to communicate.

NOTE: This step is not required and some designers may choose to send the required information on the POST JOURNAL BOD without synchronizing before hand. There are several ways to perform exception handling in this case and they will be described in the next section.

2) The second business event in the integration scenario is the posting process itself. this process flows in the direction of the general ledger component from the sub ledger components based any number of possible business event. Some of these possible business events are described in the introduction to this scenario.

1.5 Exception Handling

This section is not intended to fully define the only acceptable methods for exception handling for this scenario. It is intended to be used as a guide by the designer to understand the intent and think through the issues the creators of this model had in mind.

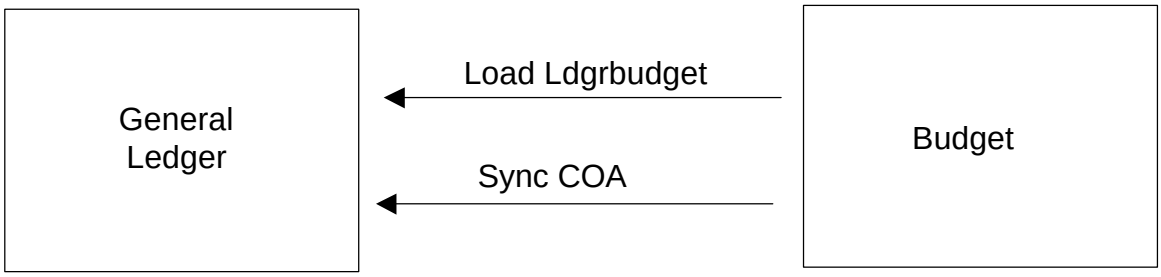
The CONFIRM BOD shown in the scenario is the most obvious method for providing an application level feedback mechanism between business software components. This CONFIRM BOD is described in detail in section three, chapter two of OAGIS, however the specific use of the CONFIRM BOD may vary significantly from scenario to scenario.

The CONFIRM BOD in this scenario is intended to be used by the general ledger component to communicate to the sub ledger that the information it sent ***was received and understood***. If the information was not received or not understood, or contained errors of any type, the financial tracking for the organization may be incomplete or incorrect.

It is strongly recommended that the CONFIRM BOD is used in this scenario to prevent this potential problem.

2.0 GENERAL LEDGER TO BUDGET

General Ledger to Budget

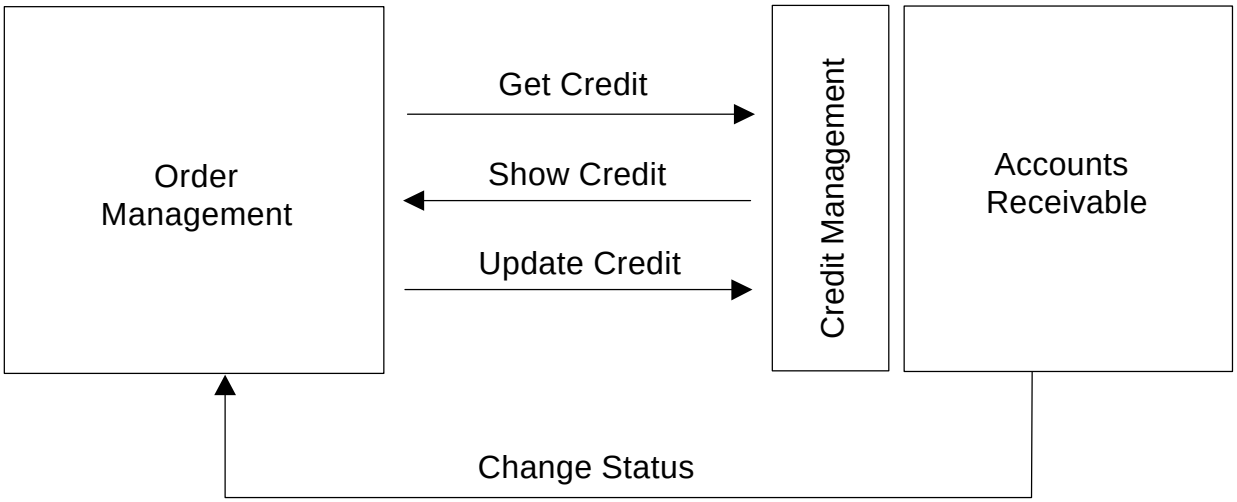


Note:

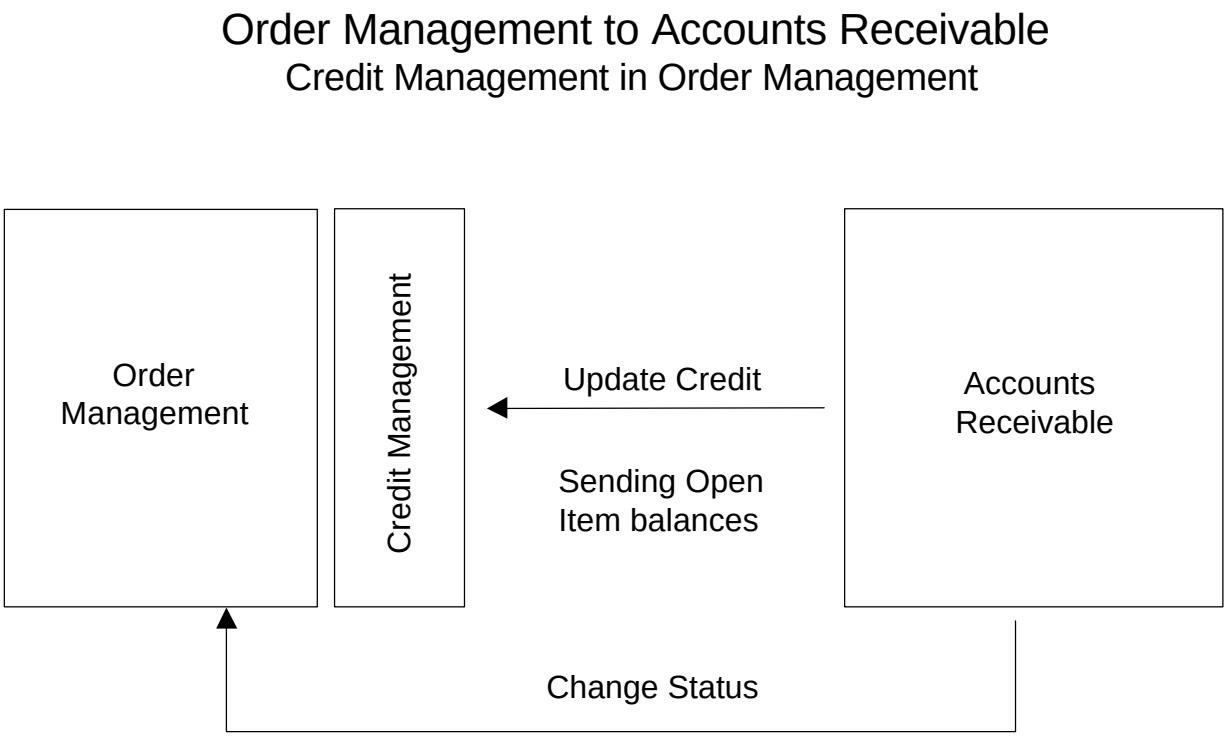
Load Revenue and Load Expense processing is defined as a human intervention point (HIP).

3.0 ORDER MANAGEMENT TO ACCOUNTS RECEIVABLE

Order Management to Accounts Receivable
Credit Management in Accounts Receivable

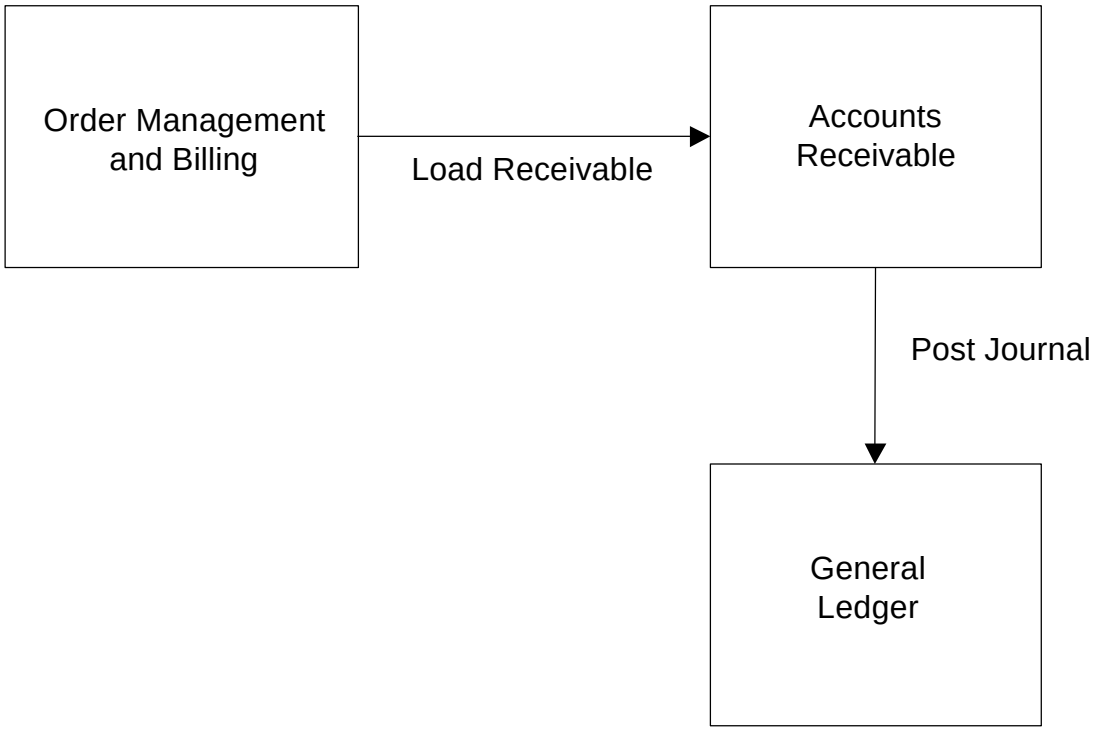


4.0 ORDER MANAGEMENT TO ACCOUNTS RECEIVABLE

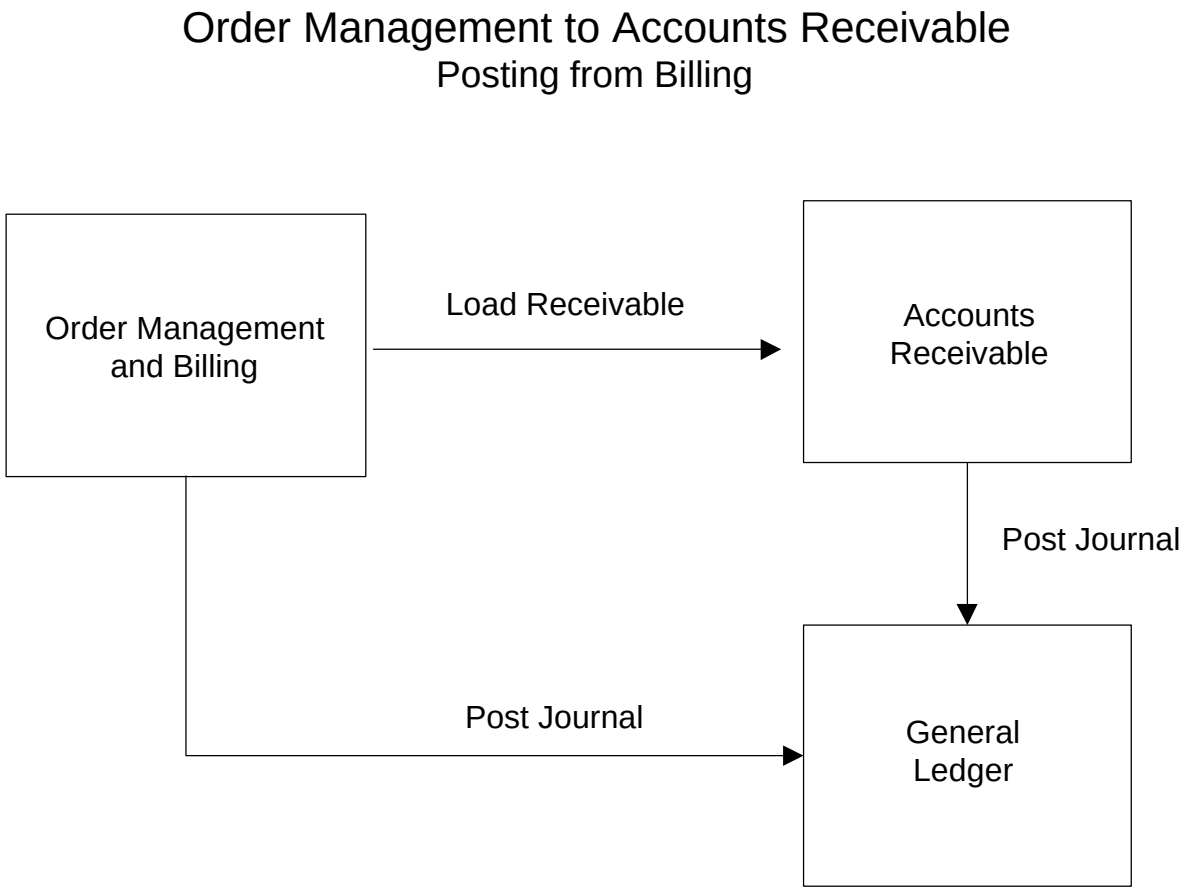


5.0 ORDER MANAGEMENT TO ACCOUNTS RECEIVABLE

Order Management to Accounts Receivable
Posting from Accounts Receivable

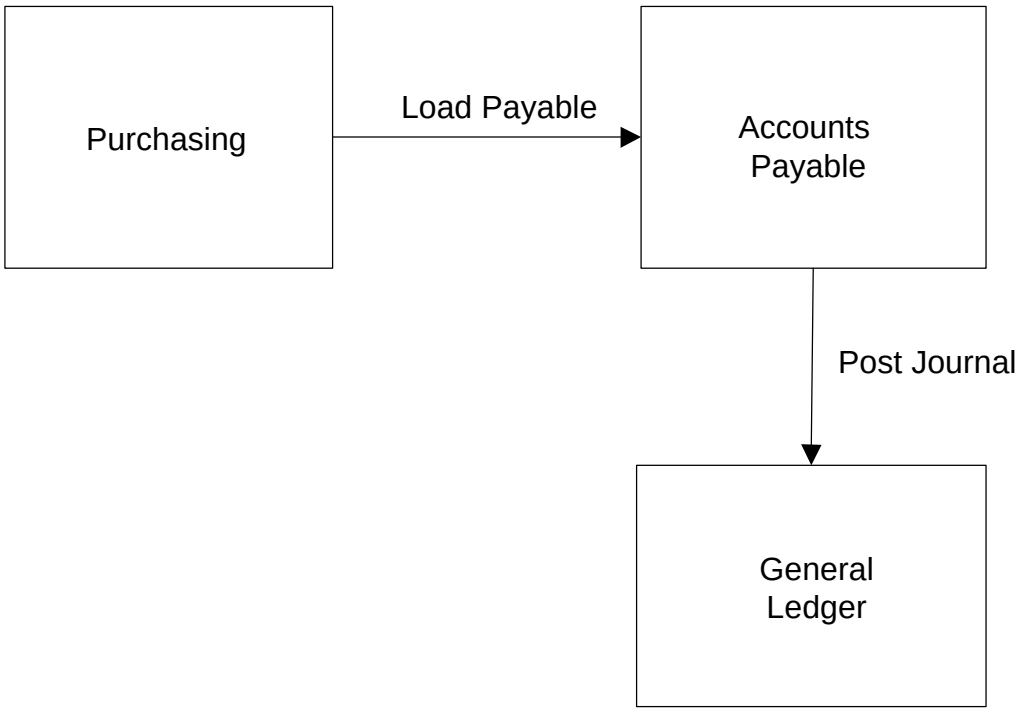


6.0 ORDER MANAGEMENT TO ACCOUNTS RECEIVABLE



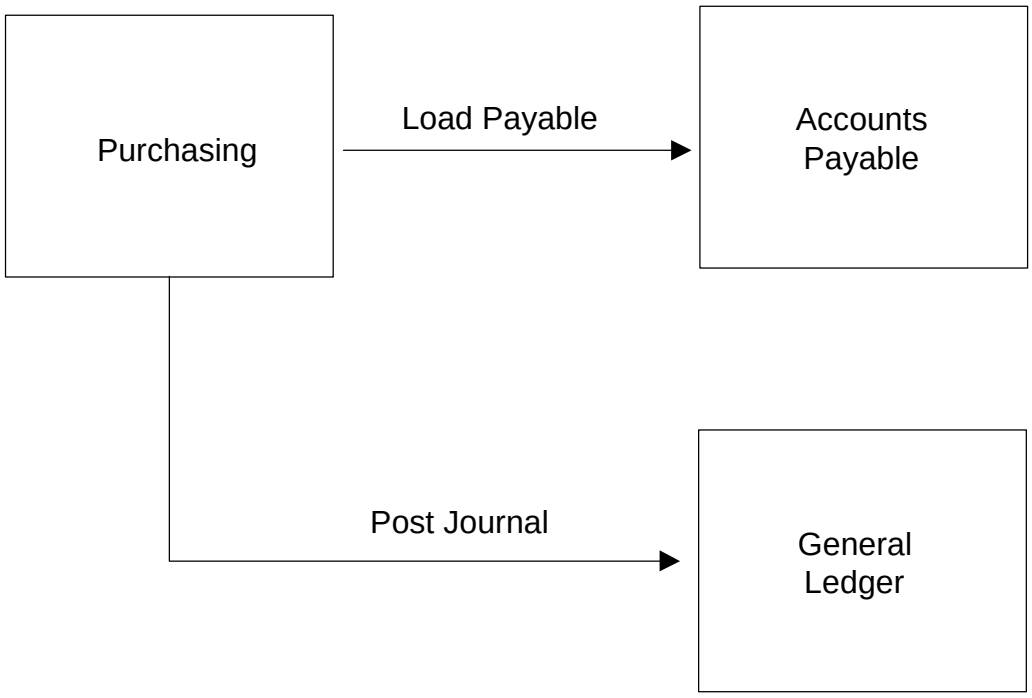
7.0 PURCHASING TO ACCOUNTS PAYABLE

Purchasing to Accounts Payable
GL Posting from Accounts Payable

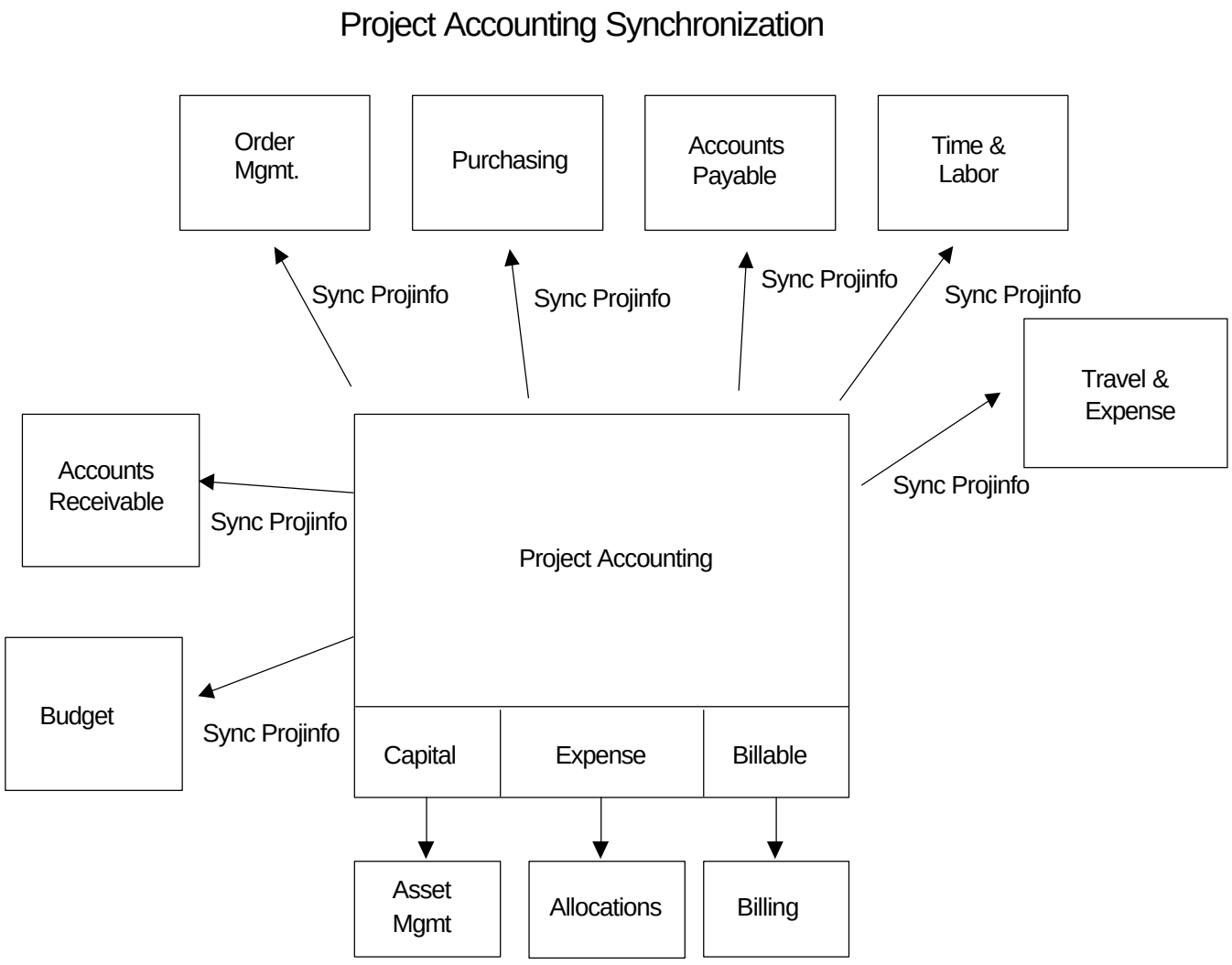


8.0 PURCHASING TO ACCOUNTS PAYABLE

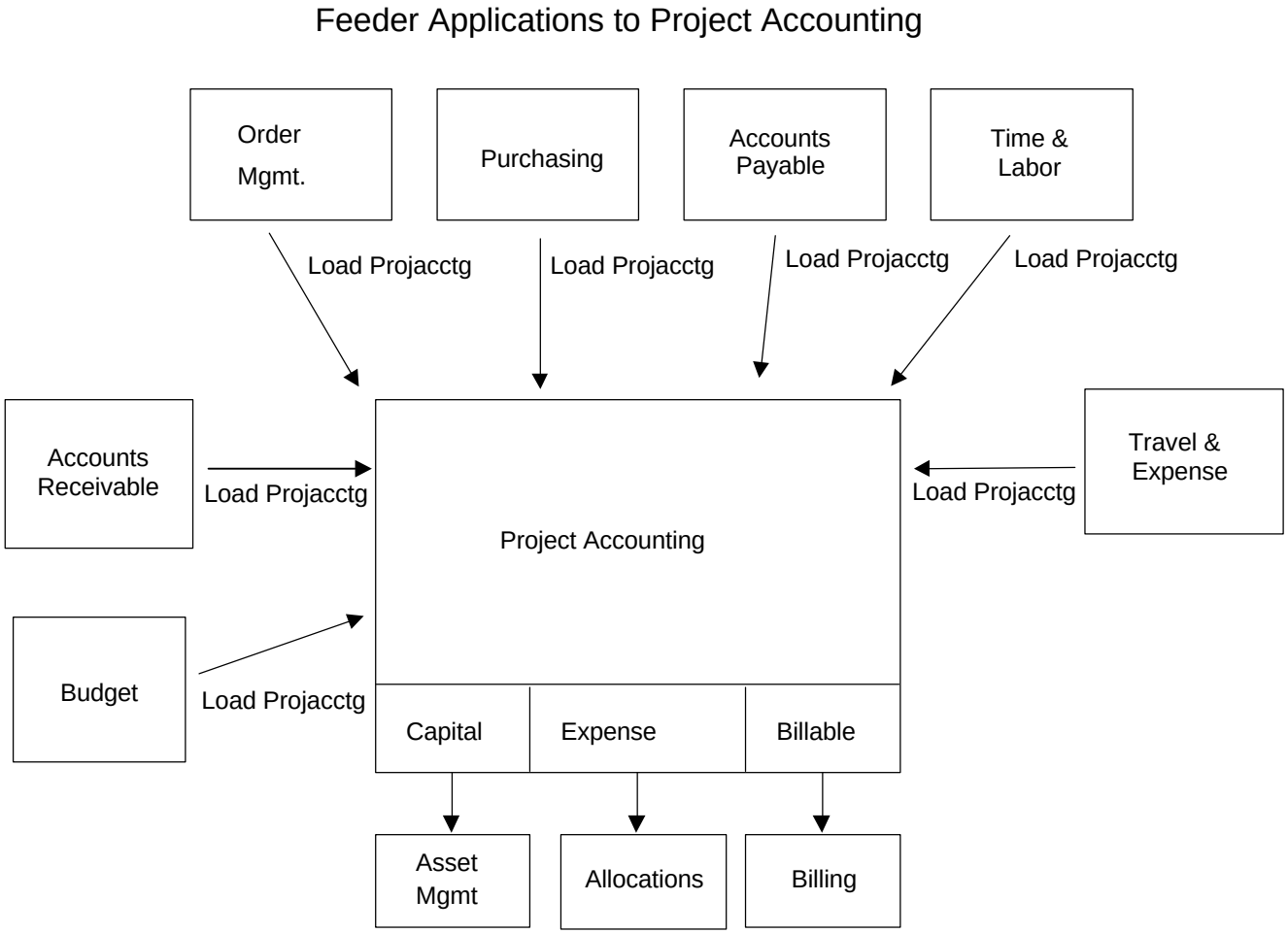
Purchasing to Accounts Payable
GL Posting from Purchasing



9.0 PROJECT ACCOUNTING SYNCHRONIZATION

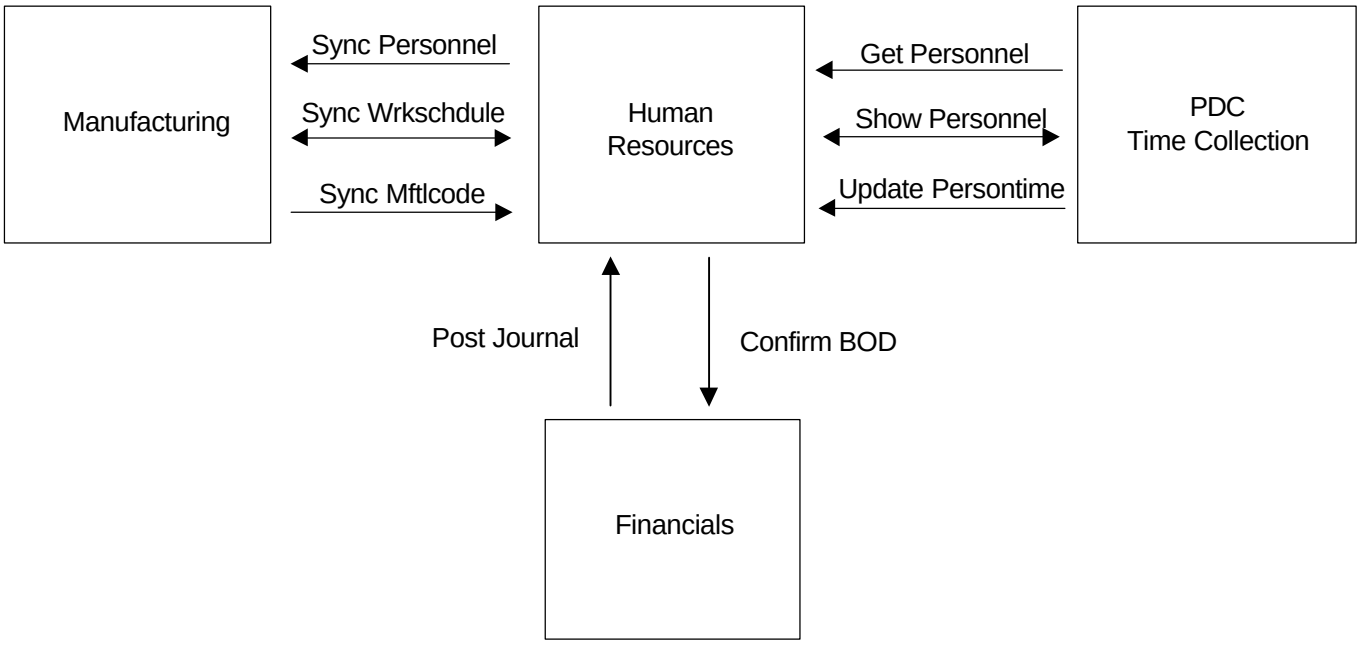


10.0 FEEDER APPLICATIONS TO PROJECT ACCOUNTING



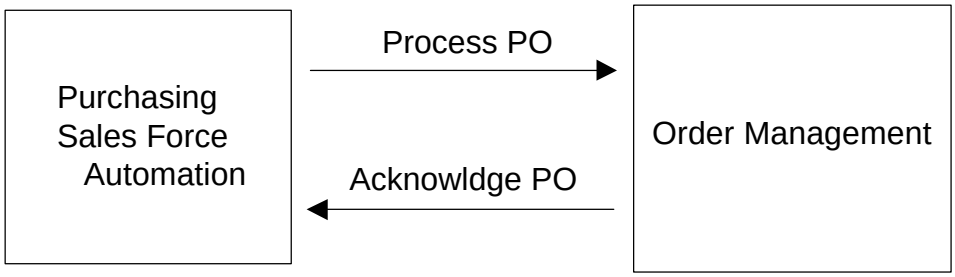
11.0 HUMAN RESOURCES INTEGRATION

Human Resources to
Manufacturing, Financials, and Plant Data Collection

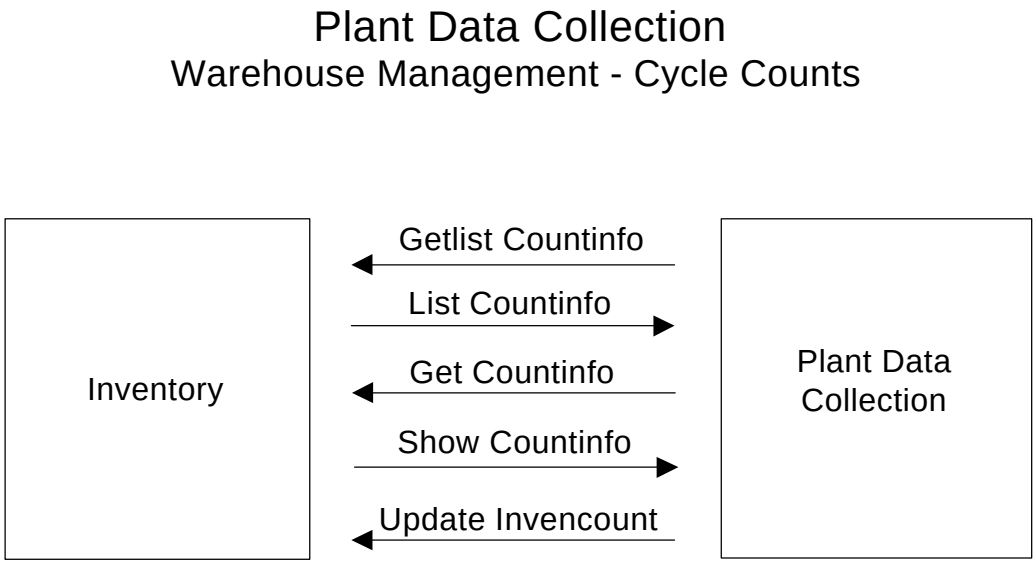


12.0 PURCHASE ORDER PROCESS

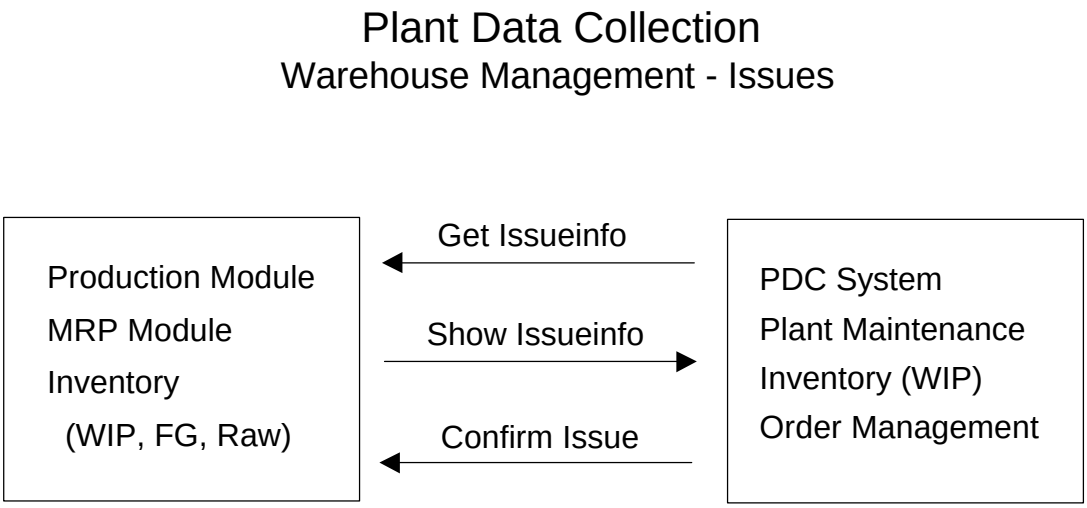
Purchasing Ordering Process



13.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (CYCLE COUNTS)

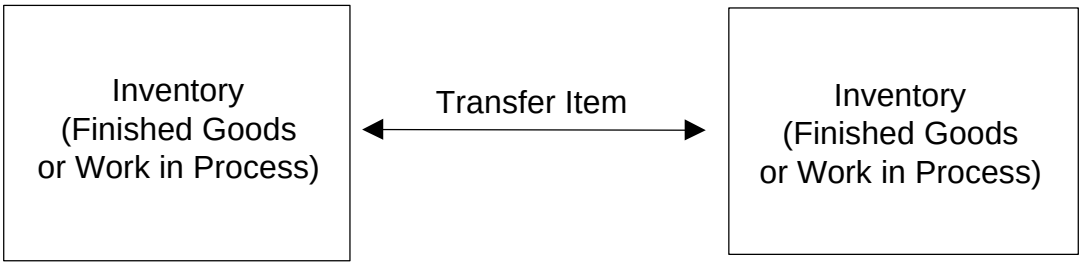


14.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (ISSUES)



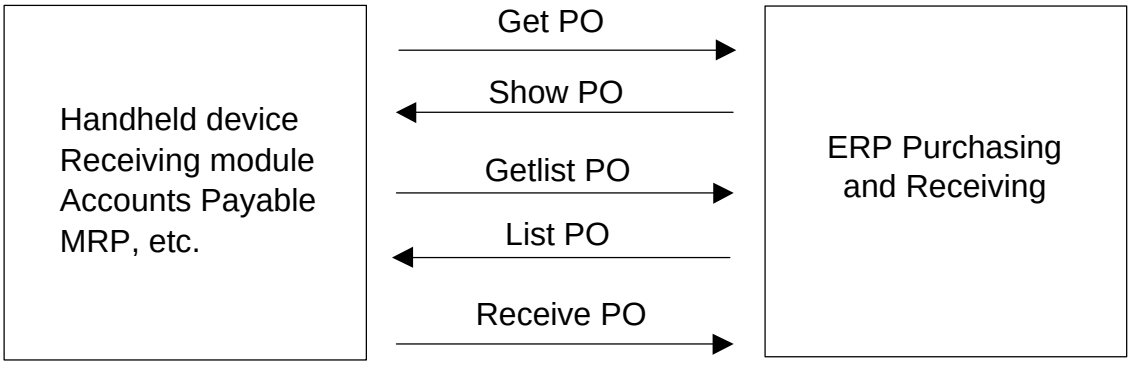
15.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (TRANSFERS)

Plant Data Collection
Warehouse Management - Transfers

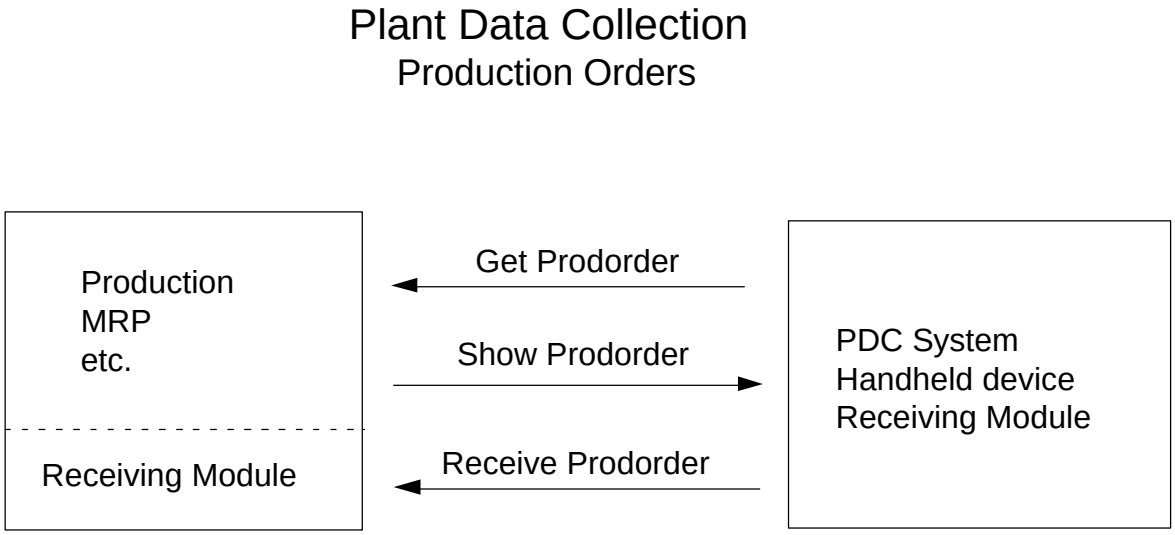


16.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (RECEIPTS)

Plant Data Collection
Warehouse Management - Receipts

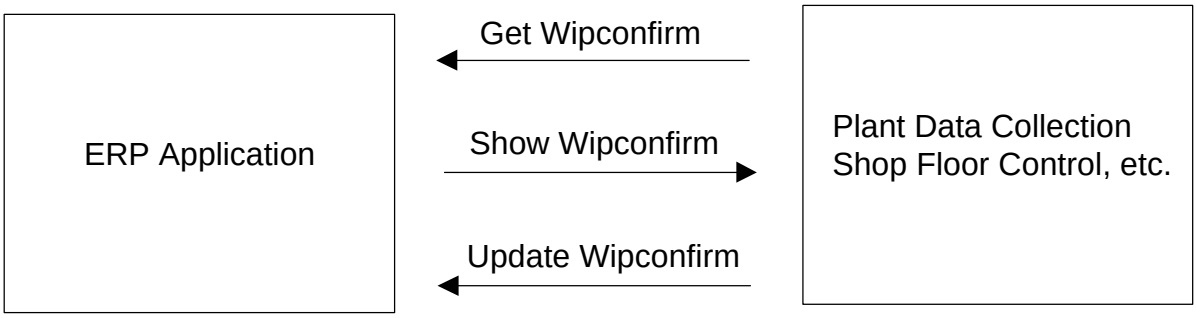


17.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (PRODUCTION ORDERS)



18.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (WORK IN PROCESS)

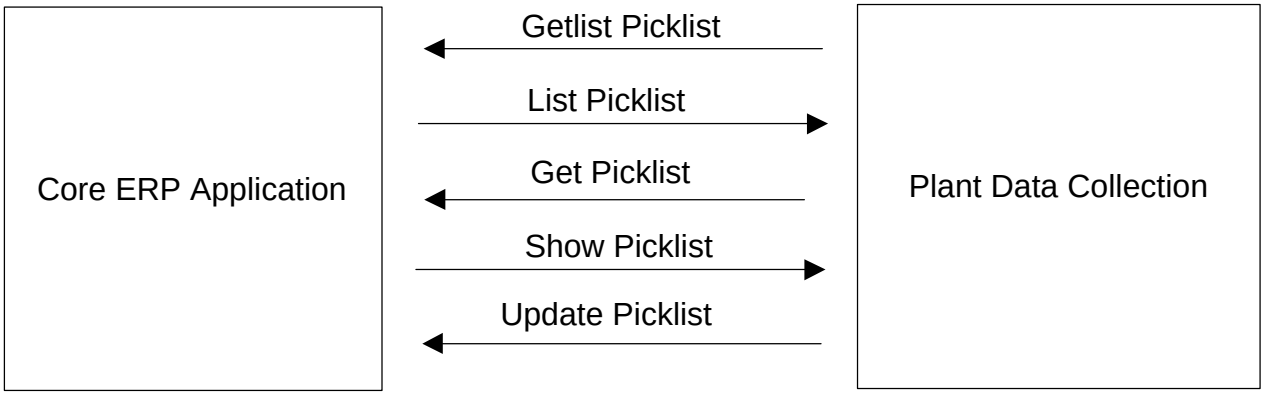
Plant Data Collection
Work in Process Confirmation Processing



19.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (SHIPPING)

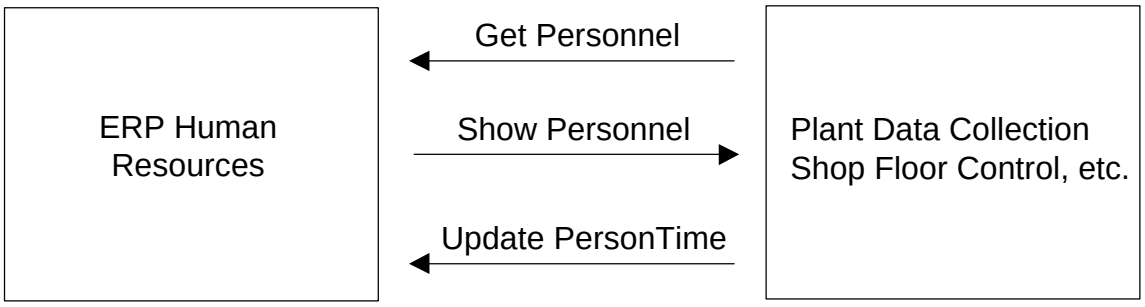
Plant Data Collection

Shipping Process

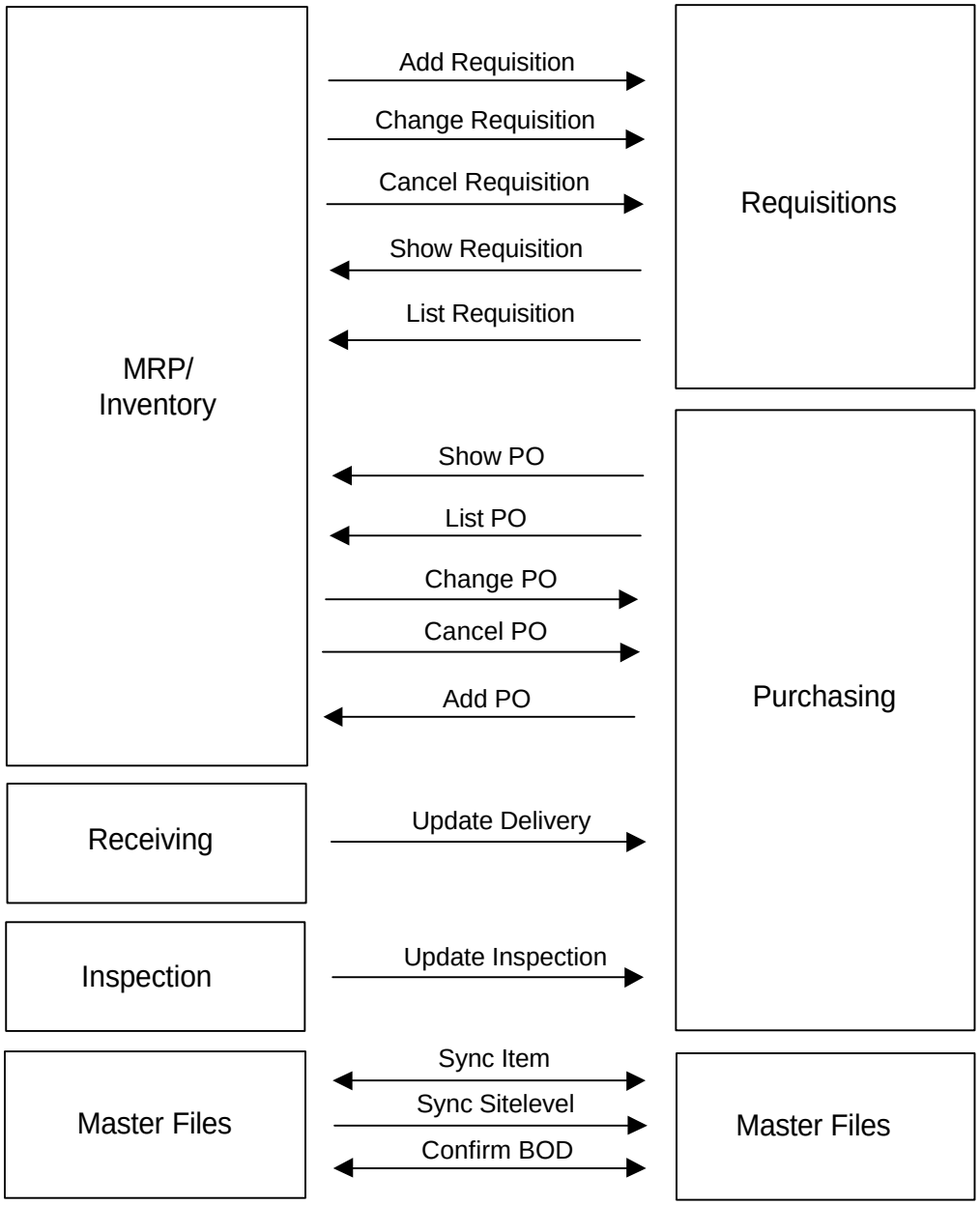


20.0 PLANT DATA COLLECTION – WAREHOUSE MANAGEMENT (TIME AND ATTENDANCE)

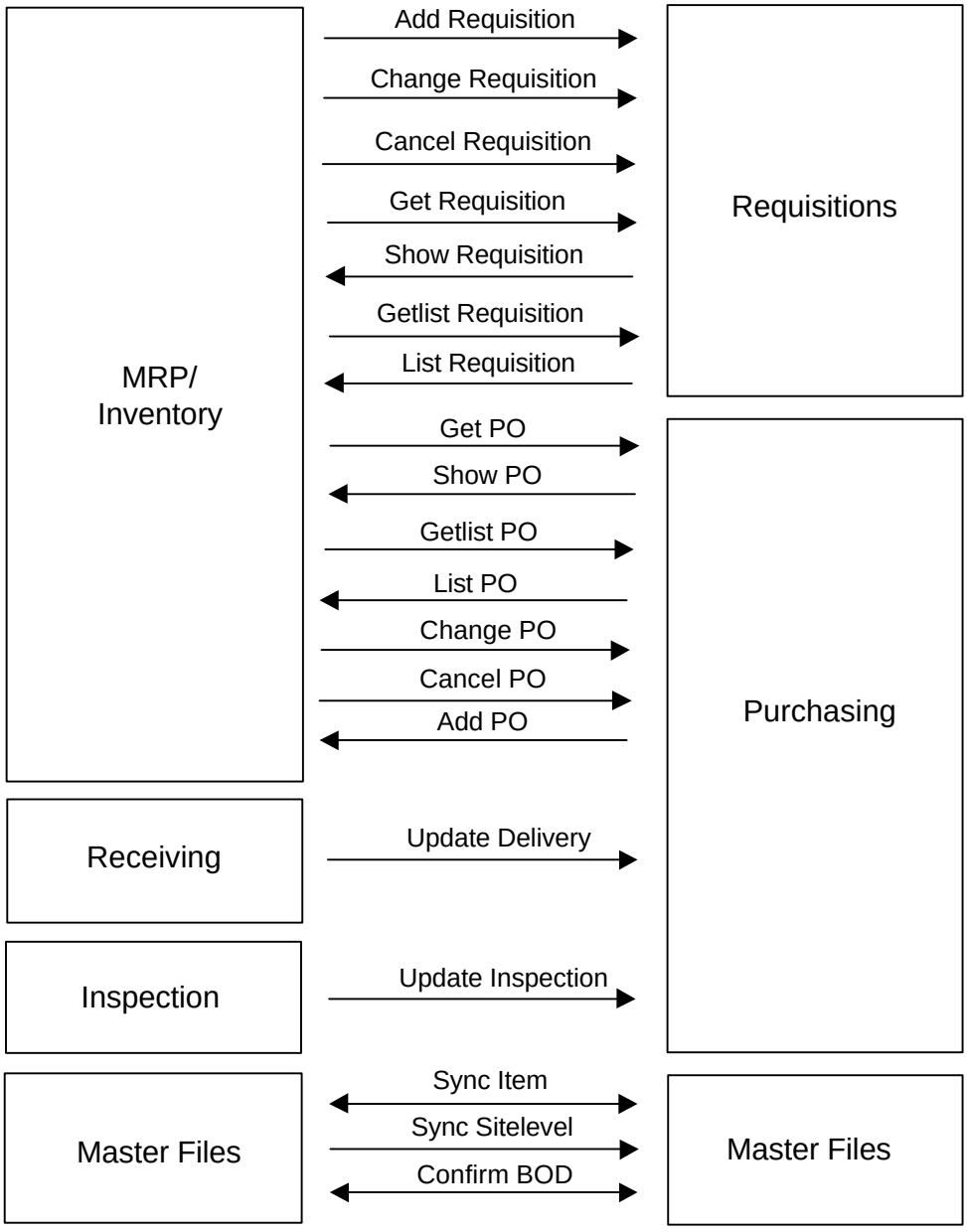
Plant Data Collection
Time and Attendance Data Gathering Process



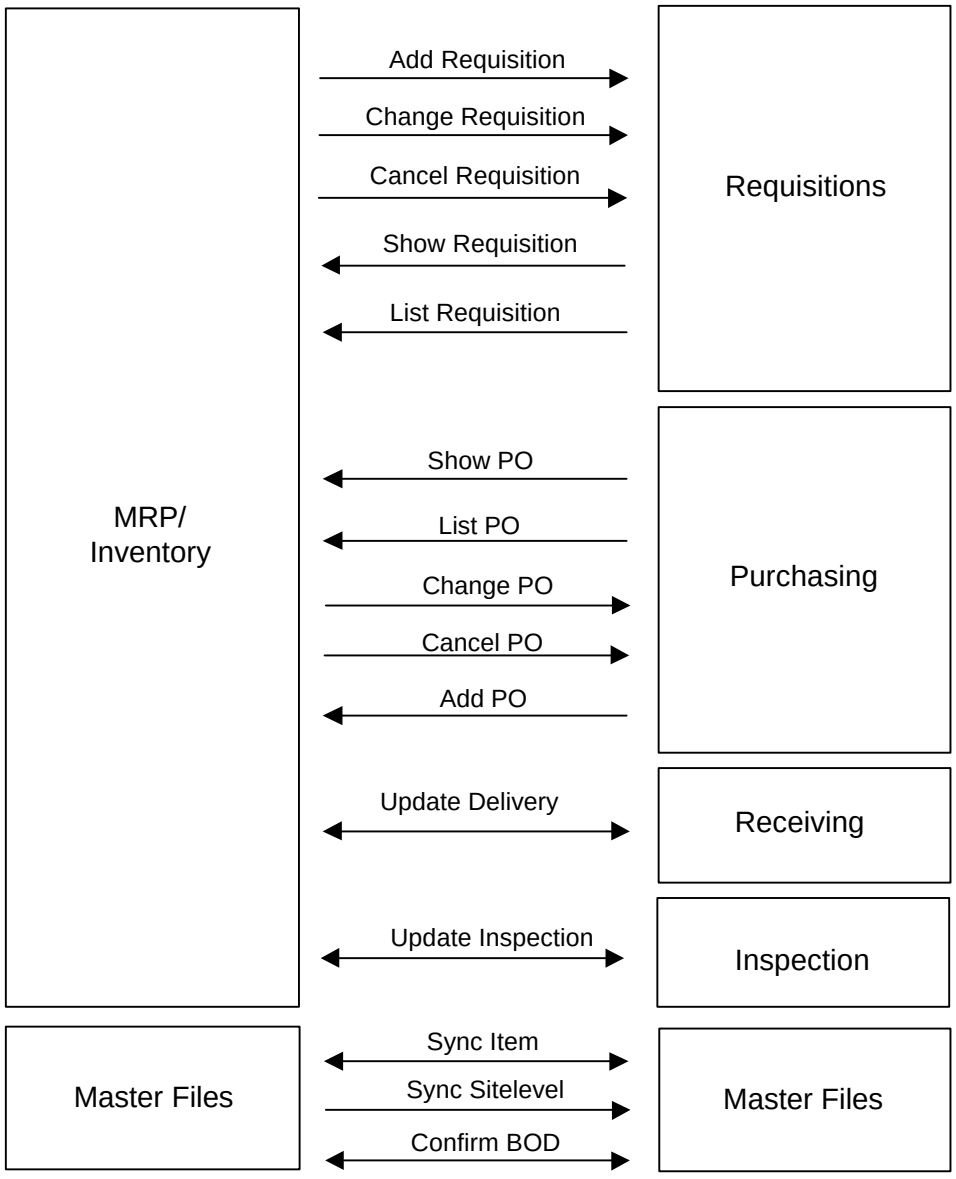
21.0 MANUFACTURING TO PURCHASING – RECEIVING AND INSPECTION IN MANUFACTURING (PUBLISH/SUBSCRIBE MODEL)



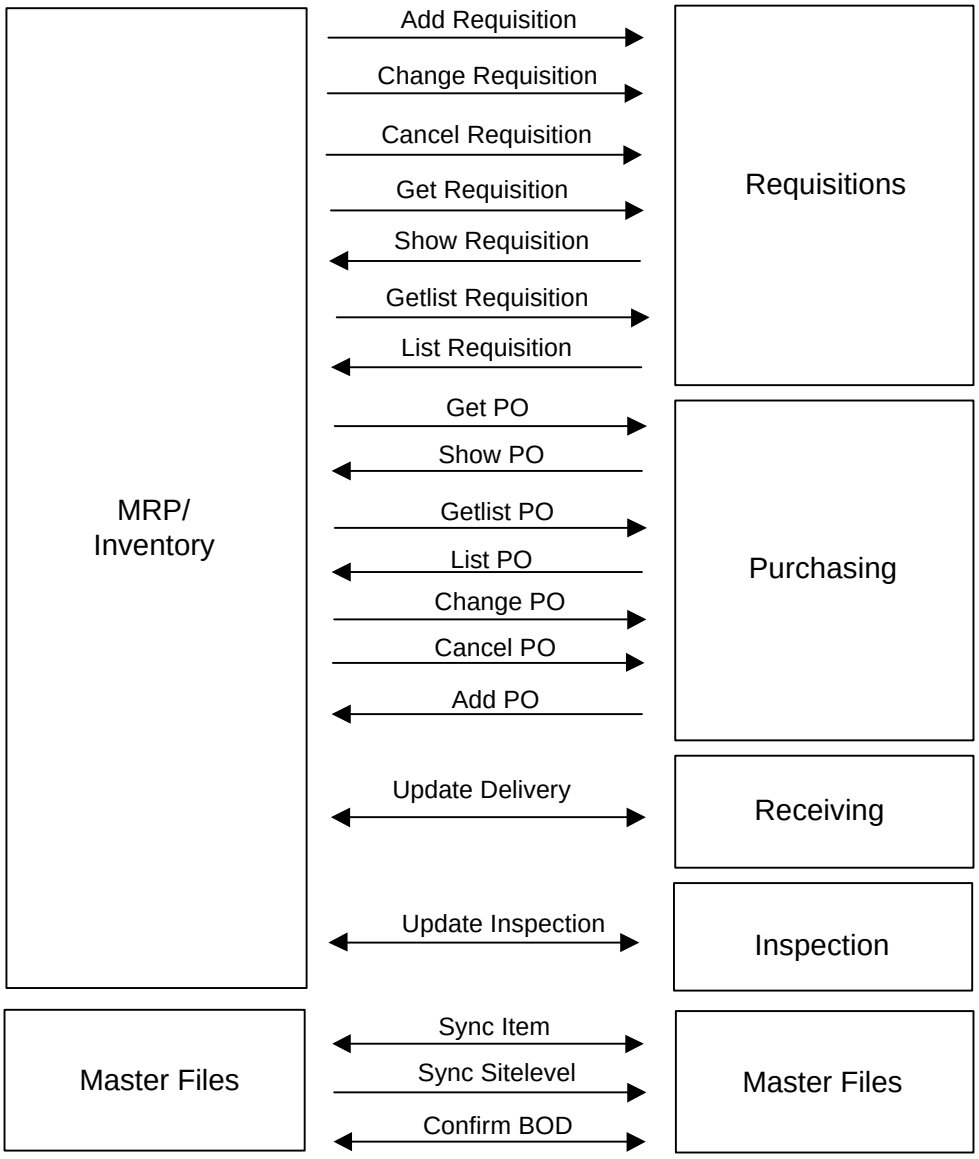
22.0 MANUFACTURING TO PURCHASING – RECEIVING AND INSPECTION IN MANUFACTURING (REQUEST/REPLAY AND PUBLISH/SUBSCRIBE)



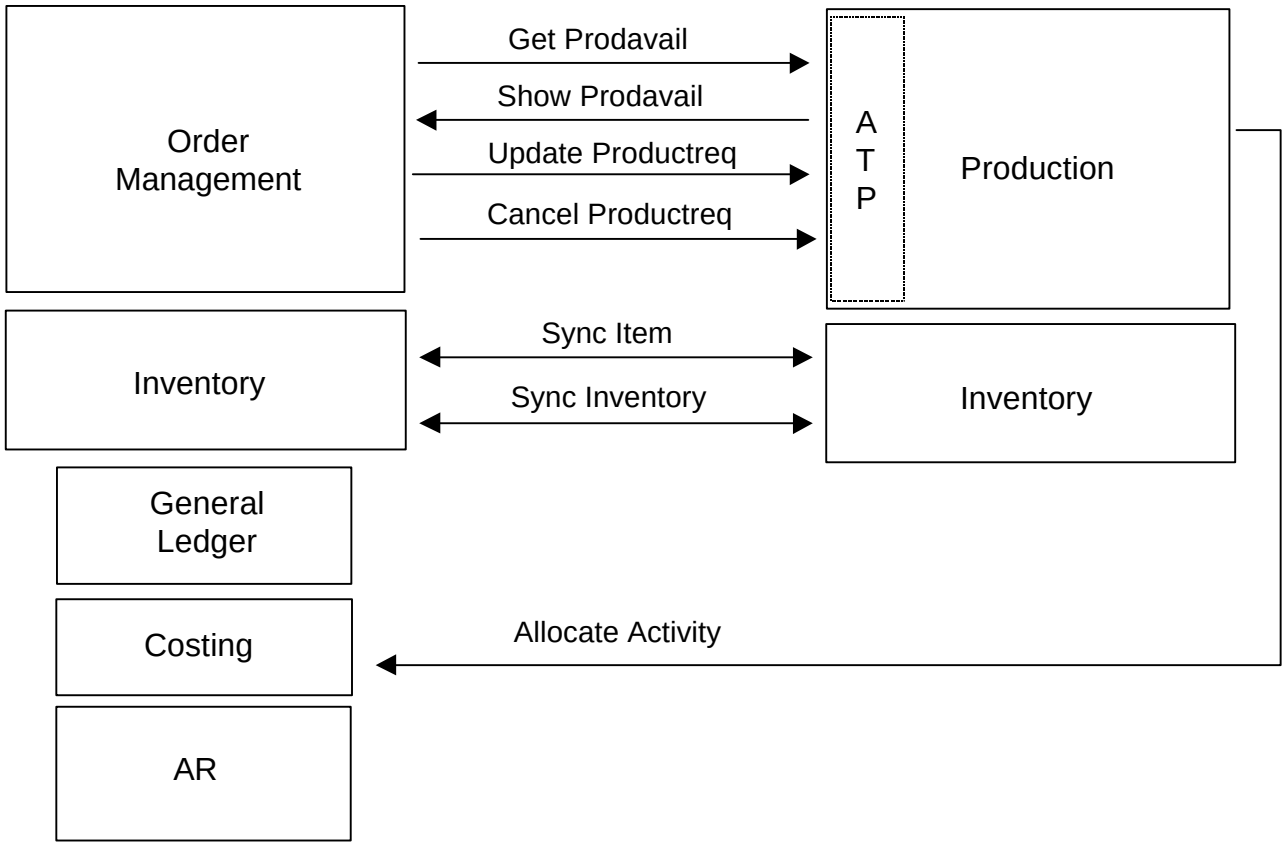
23.0 MANUFACTURING TO PURCHASING – RECEIVING AND INSPECTION IN PURCHASING (PUBLISH/SUBSCRIBE)



24.0 MANUFACTURING TO PURCHASING – RECEIVING AND INSPECTION IN PURCHASING (REQUEST/REPLY AND PUBLISH/SUBSCRIBE)

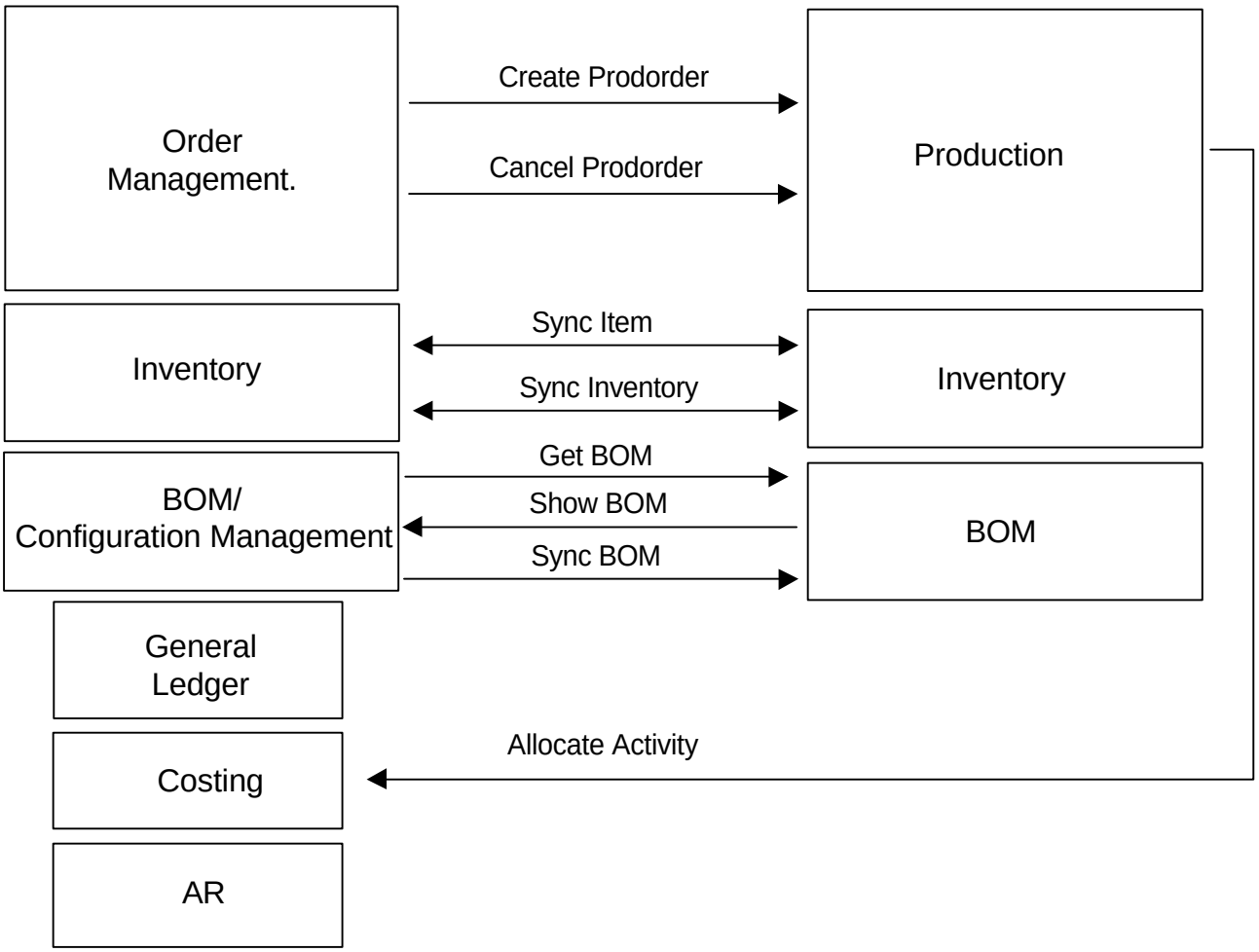


25.0 MANUFACTURING TO ORDER
MANAGEMENT – FINANCIALS WITH LOGISTICS, (MAKE TO ORDER, BUILD TO ORDER)



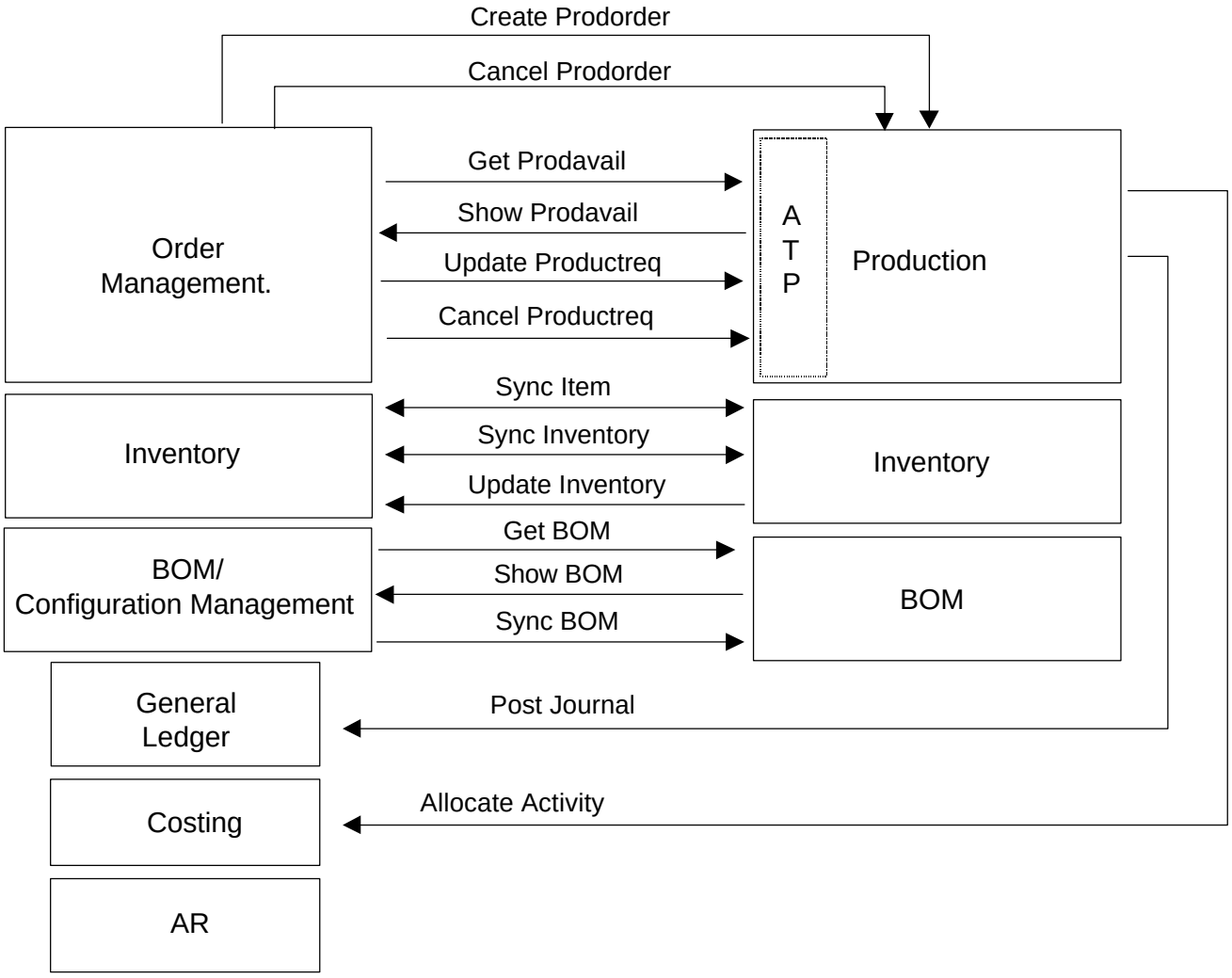
26.0 MANUFACTURING TO ORDER

MANAGEMENT – FINANCIALS WITH LOGISTICS, (ENGINEER TO ORDER, CONFIGURE TO ORDER)



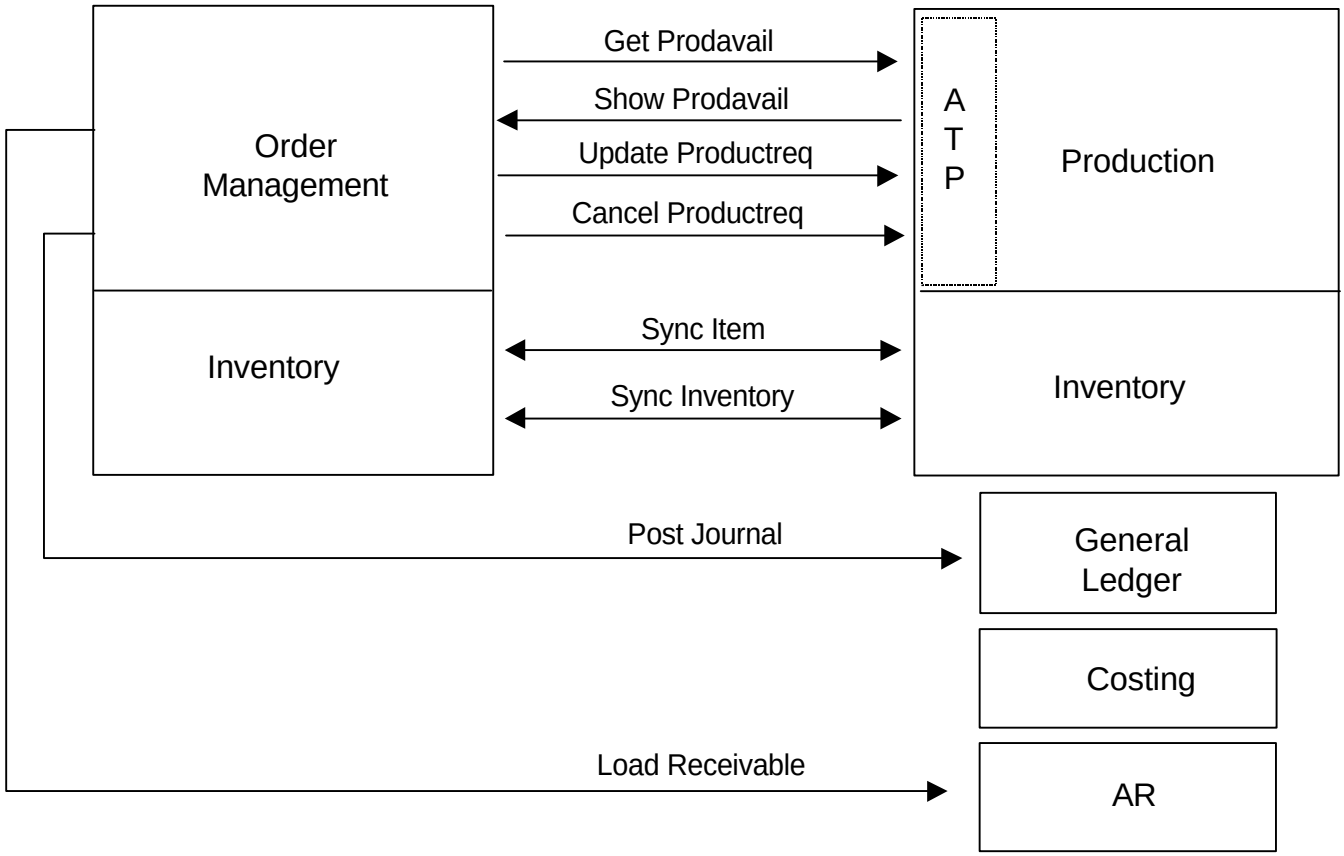
27.0 MANUFACTURING TO ORDER

MANAGEMENT – FINANCIALS WITH LOGISTICS, (MIXED MODE MANUFACTURING)



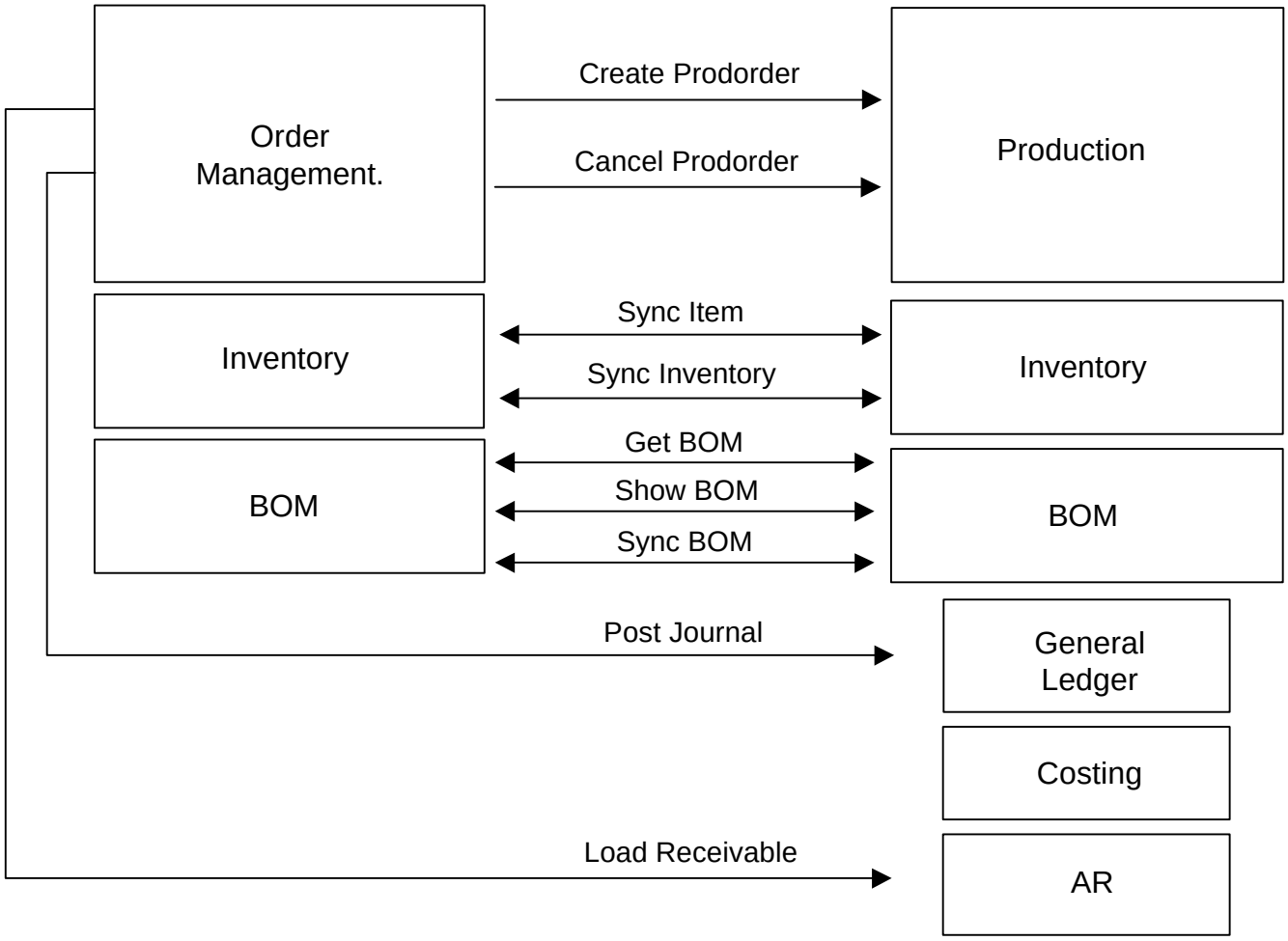
28.0 MANUFACTURING TO ORDER

MANAGEMENT – FINANCIALS WITH MANUFACTURING, (MAKE TO ORDER, BUILD TO ORDER)



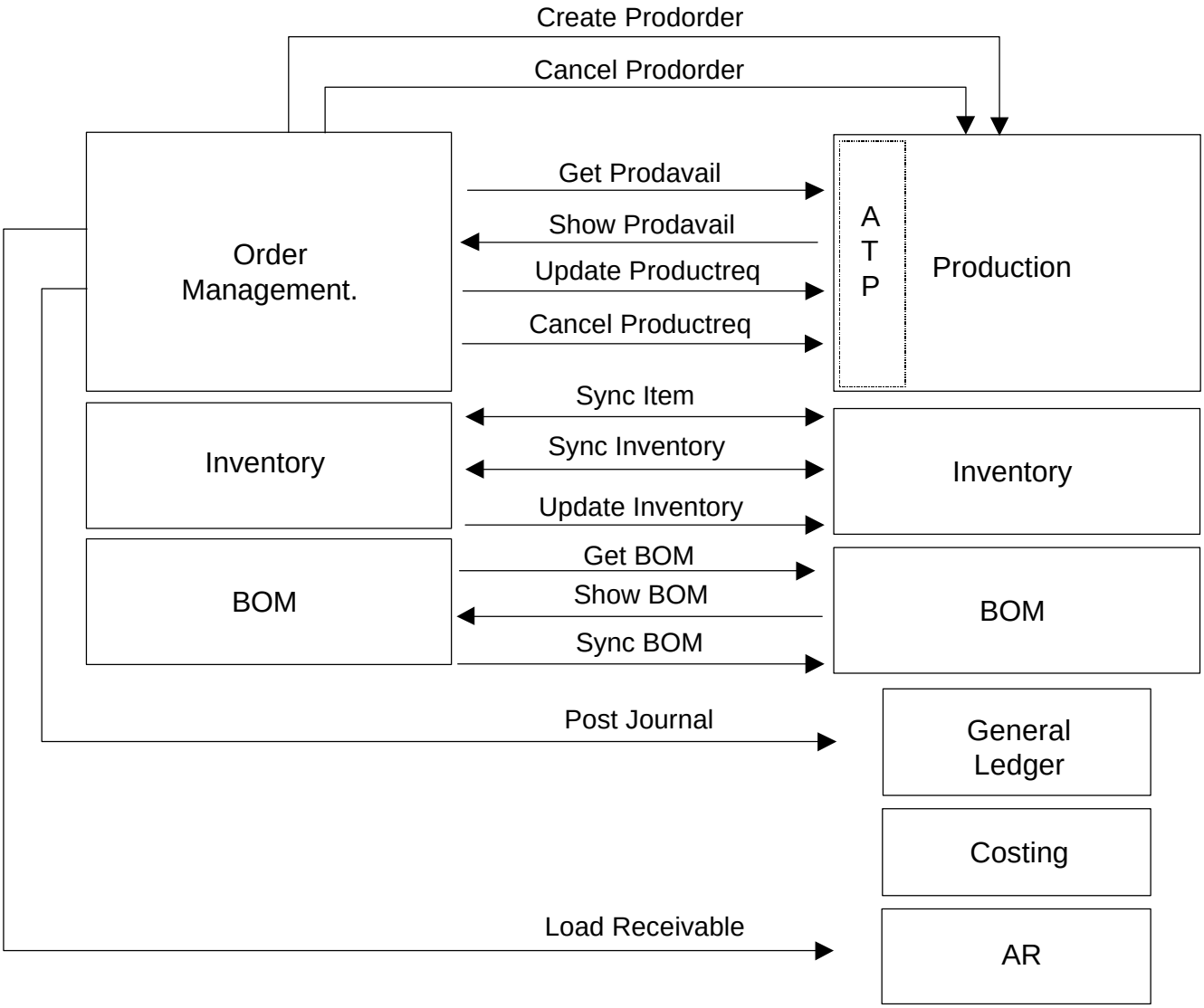
29.0 MANUFACTURING TO ORDER

MANAGEMENT – FINANCIALS WITH MANUFACTURING, (ENGINEER TO ORDER, CONFIGURE TO ORDER)

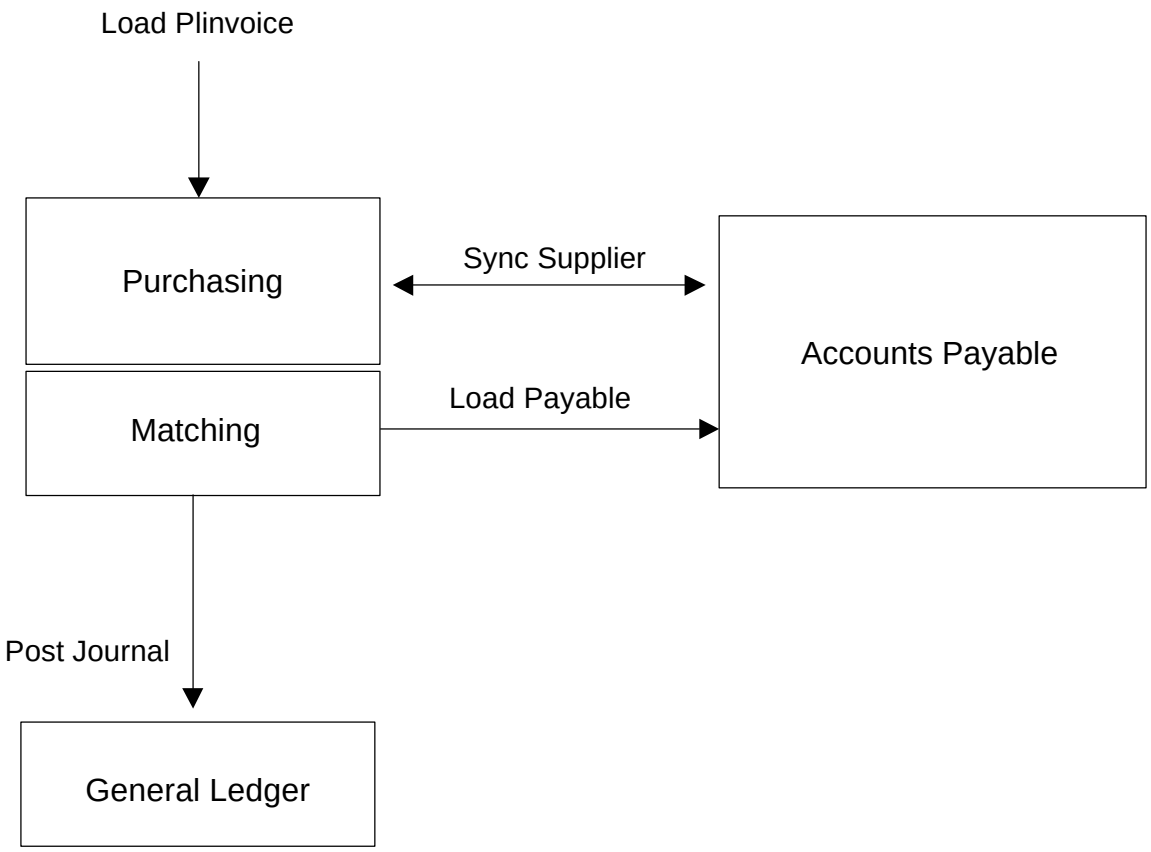


30.0 MANUFACTURING TO ORDER

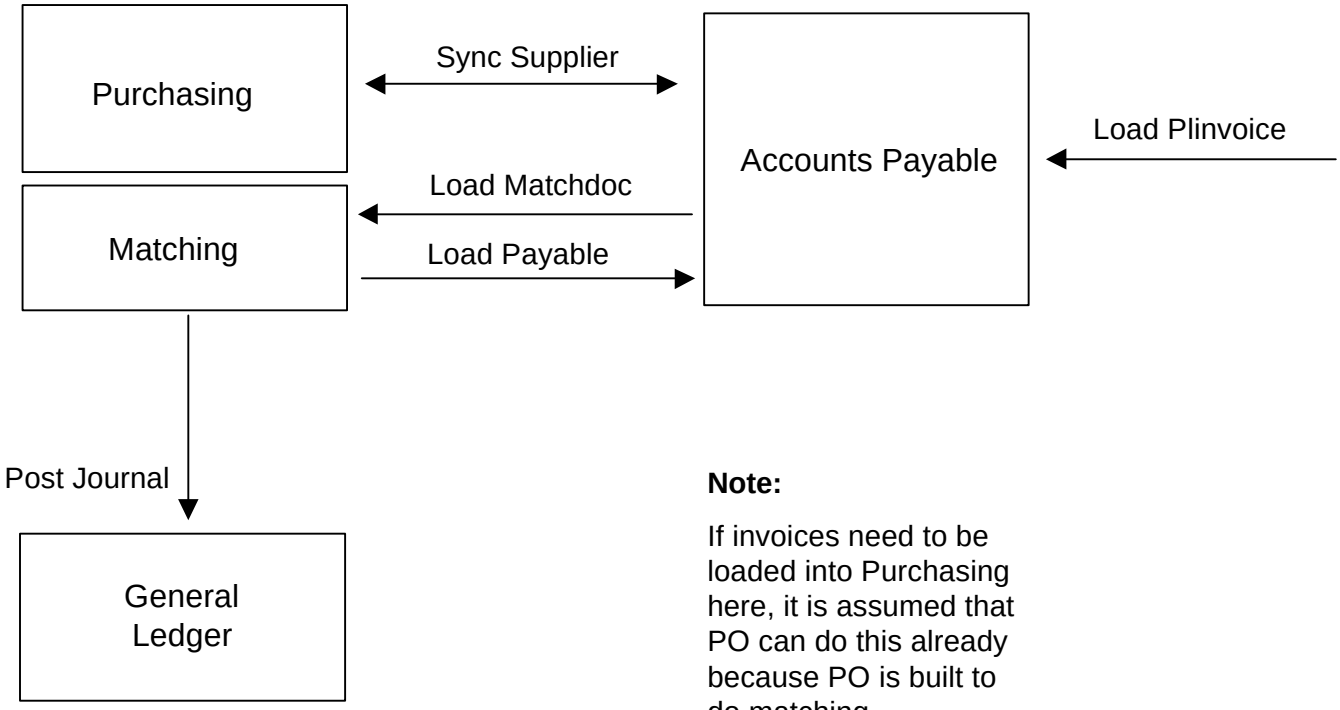
MANAGEMENT – FINANCIALS WITH MANUFACTURING, (MIXED MODE MANUFACTURING)



31.0 INVOICE MATCHING, MATCHING IN PURCHASING, INVOICES ENTERED IN PURCHASING



32.0 INVOICE MATCHING,
MATCHING IN PURCHASING, INVOICES ENTERED IN ACCOUNTS PAYABLE (PUBLISH/SUBSCRIBE)

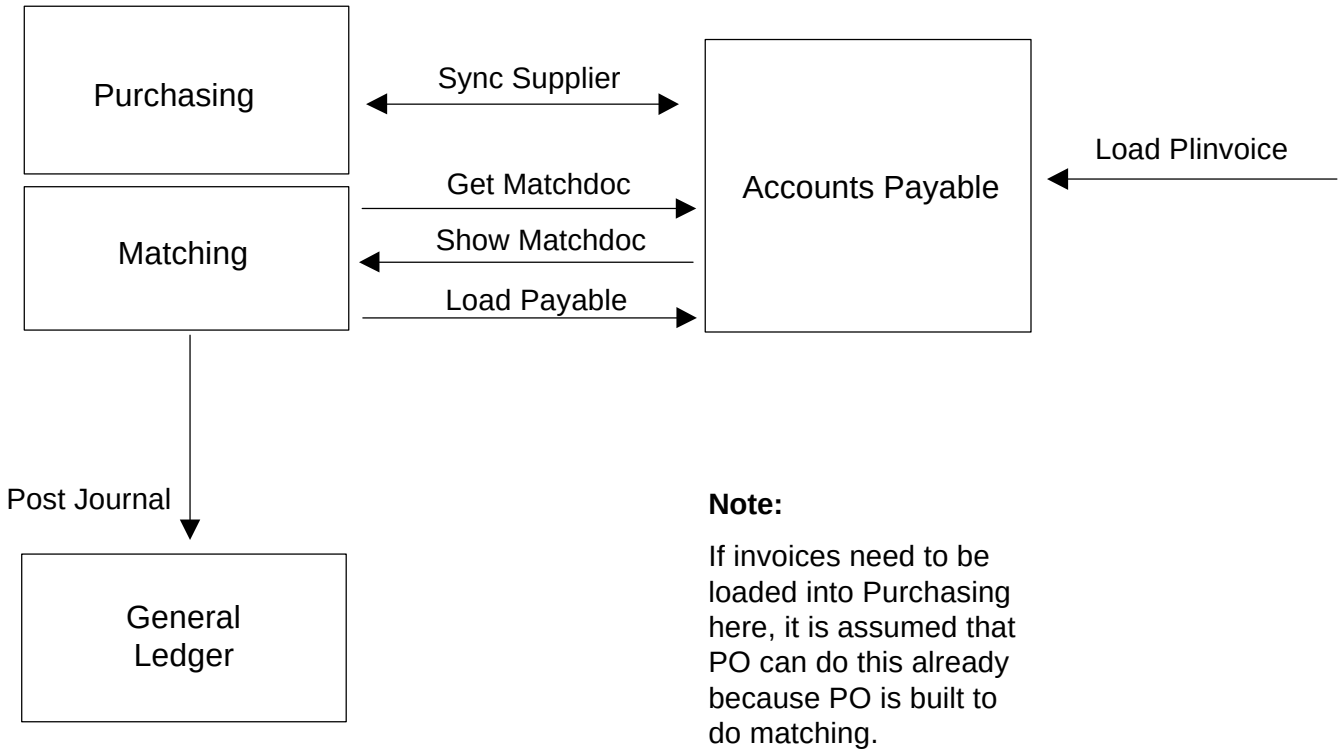


Note:

If invoices need to be loaded into Purchasing here, it is assumed that PO can do this already because PO is built to do matching.

33.0 INVOICE MATCHING, MATCHING IN PURCHASING, INVOICES ENTERED IN ACCOUNTS PAYABLE (REQUEST/REPLY)

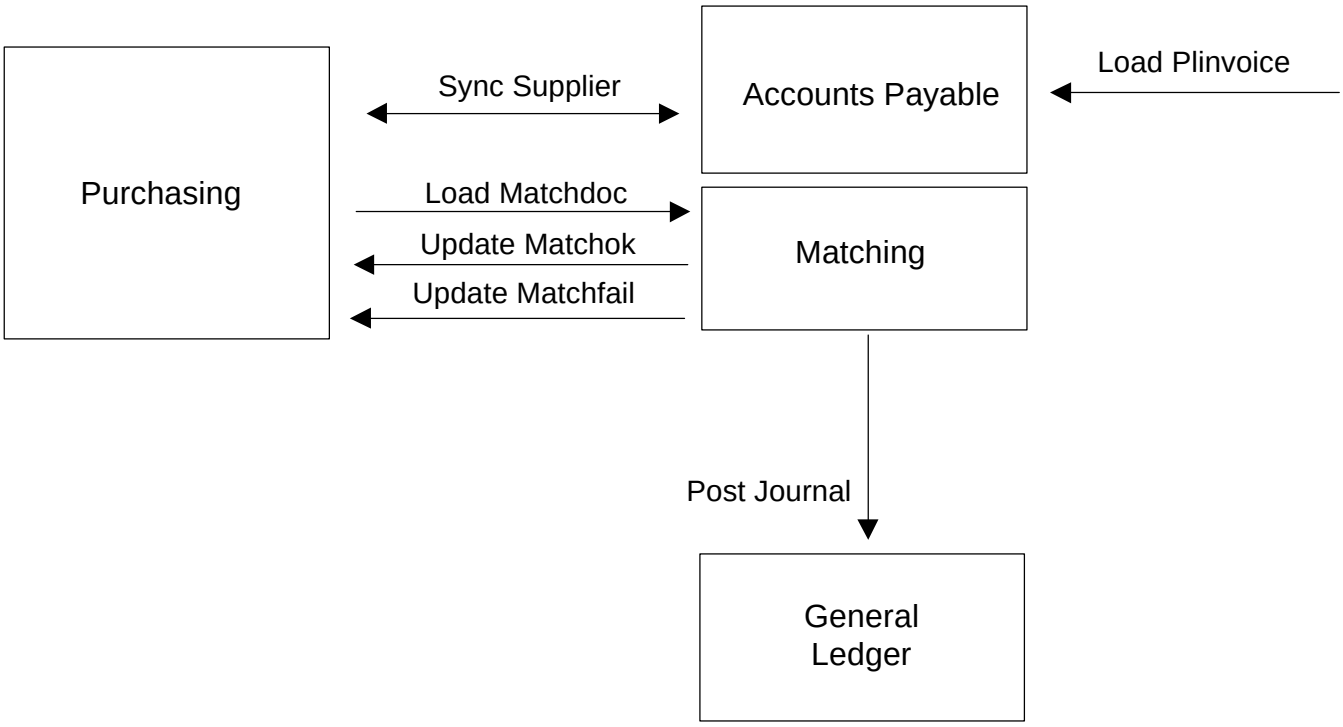
Request/Reply, or Pull Model



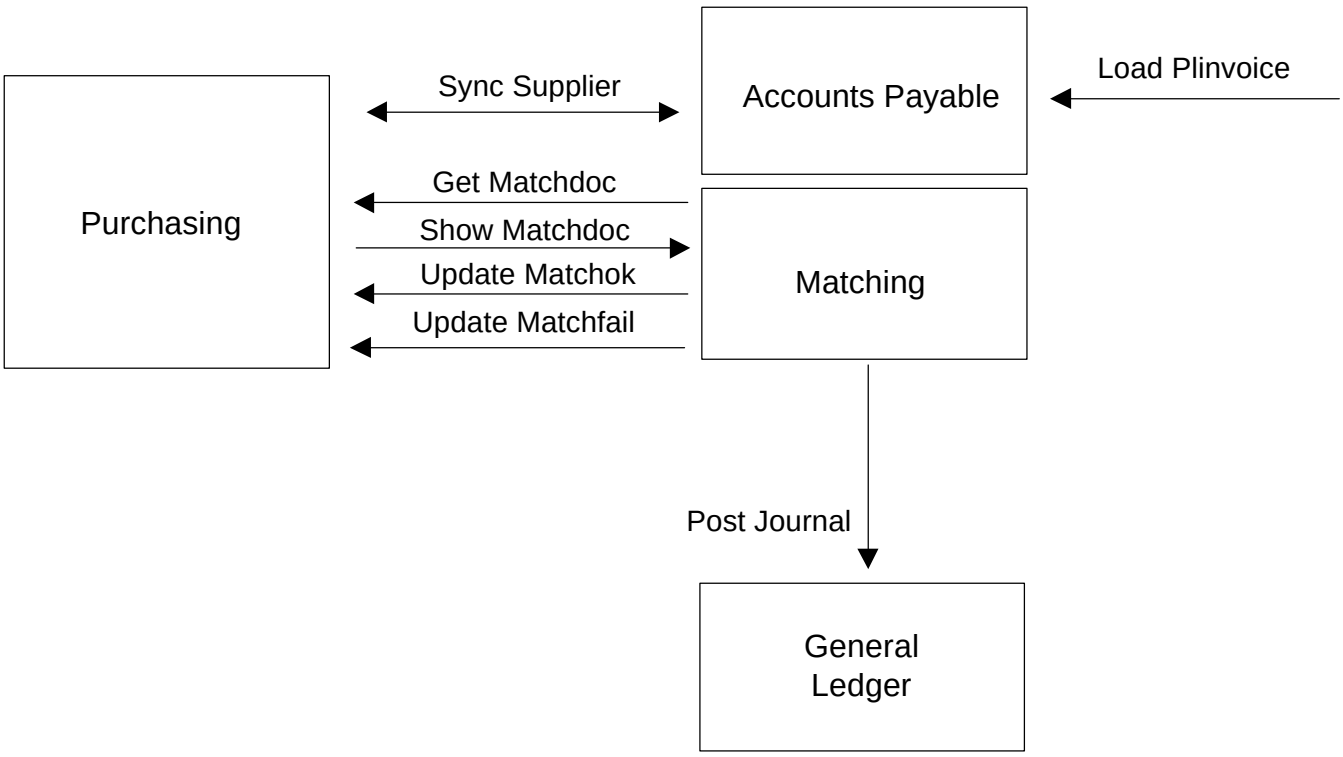
Note:

If invoices need to be loaded into Purchasing here, it is assumed that PO can do this already because PO is built to do matching.

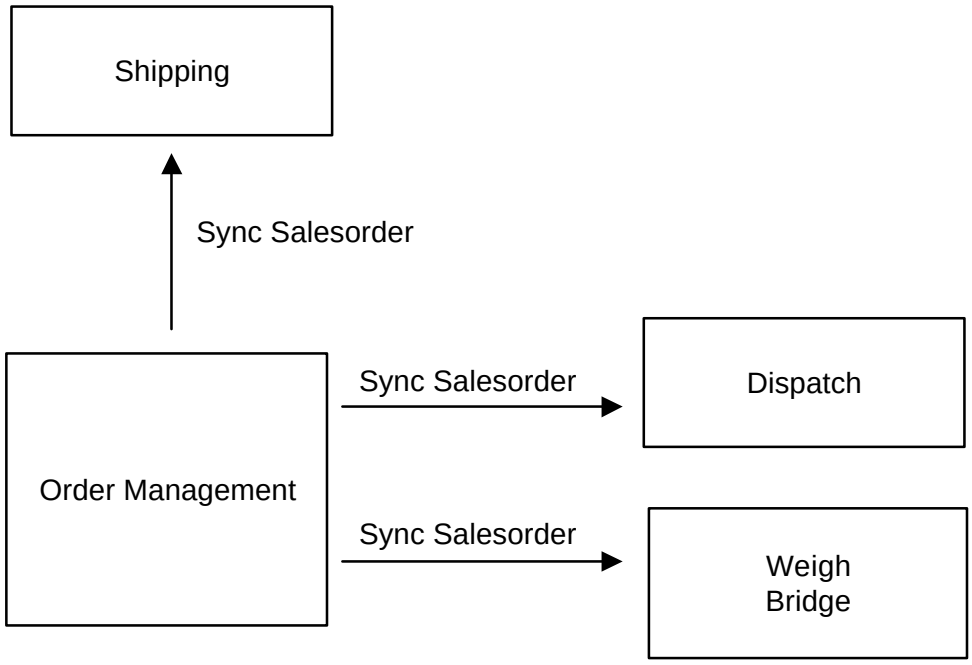
34.0 INVOICE MATCHING,
MATCHING IN ACCOUNTS PAYABLE, INVOICES ENTERED IN ACCOUNTS PAYABLE (PUBLISH/SUBSCRIBE)



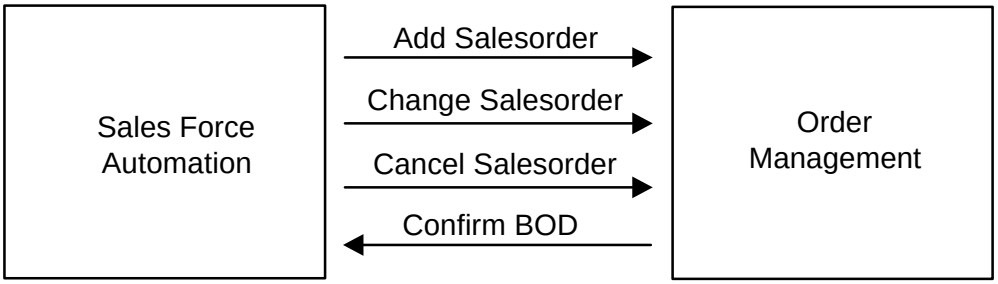
35.0 INVOICE MATCHING,
MATCHING IN ACCOUNTS PAYABLE, (REQUEST/REPLY)



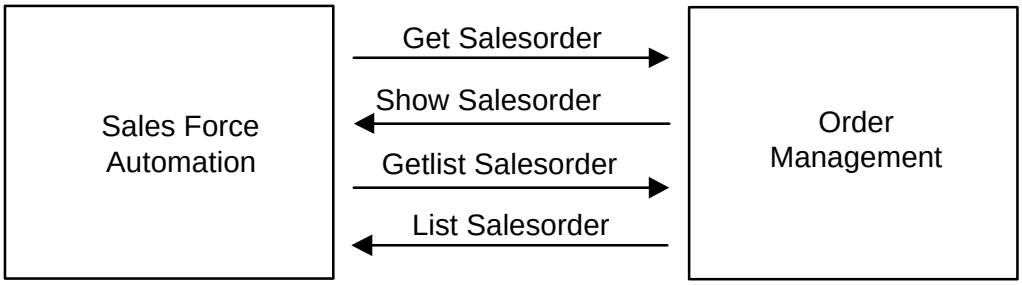
36.0 SYNCHRONIZE SALES ORDERS FOR SHIPPING



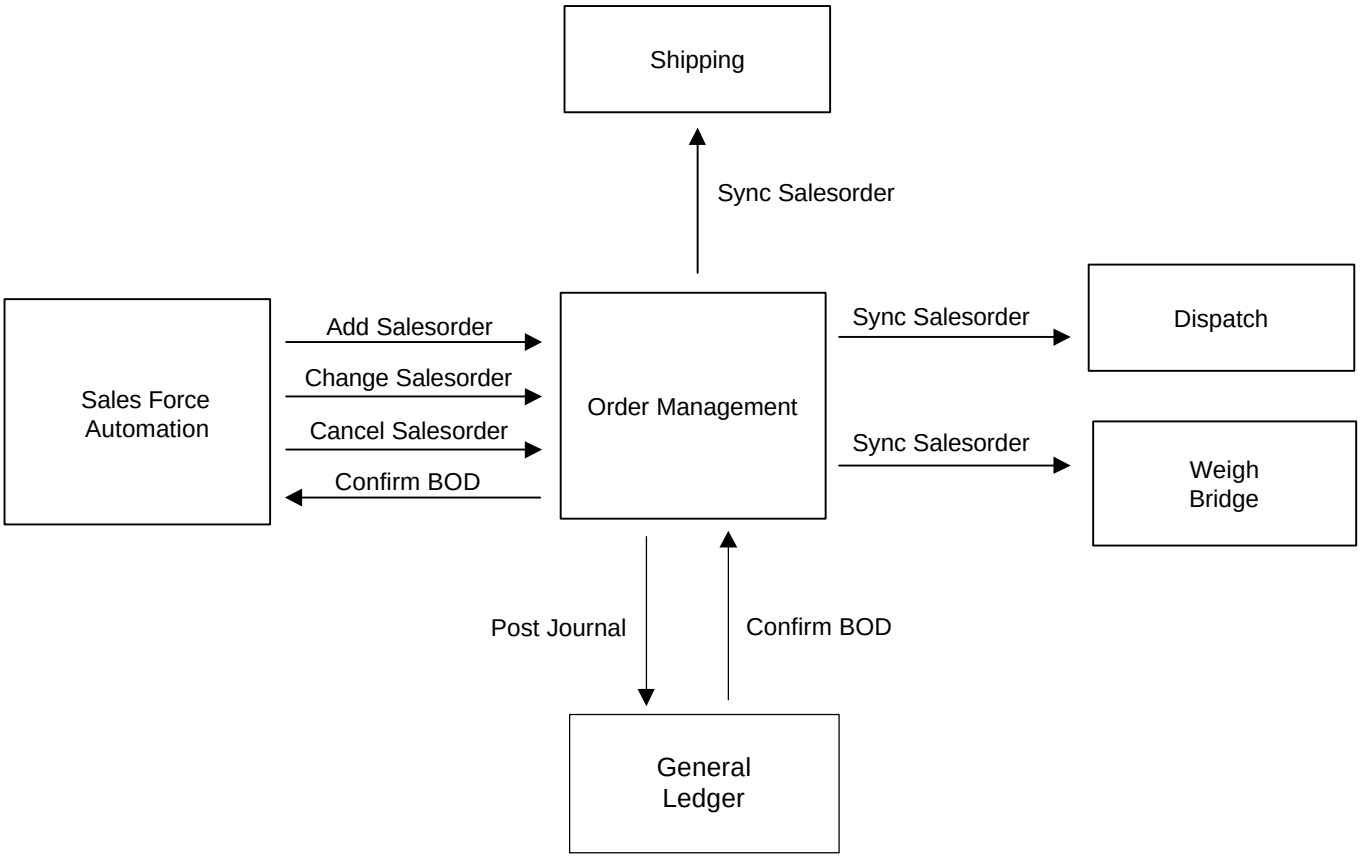
37.0 SALES FORCE AUTOMATION TO ORDER MANAGEMENT, UPDATING ORDERS IN ORDER MANAGEMENT



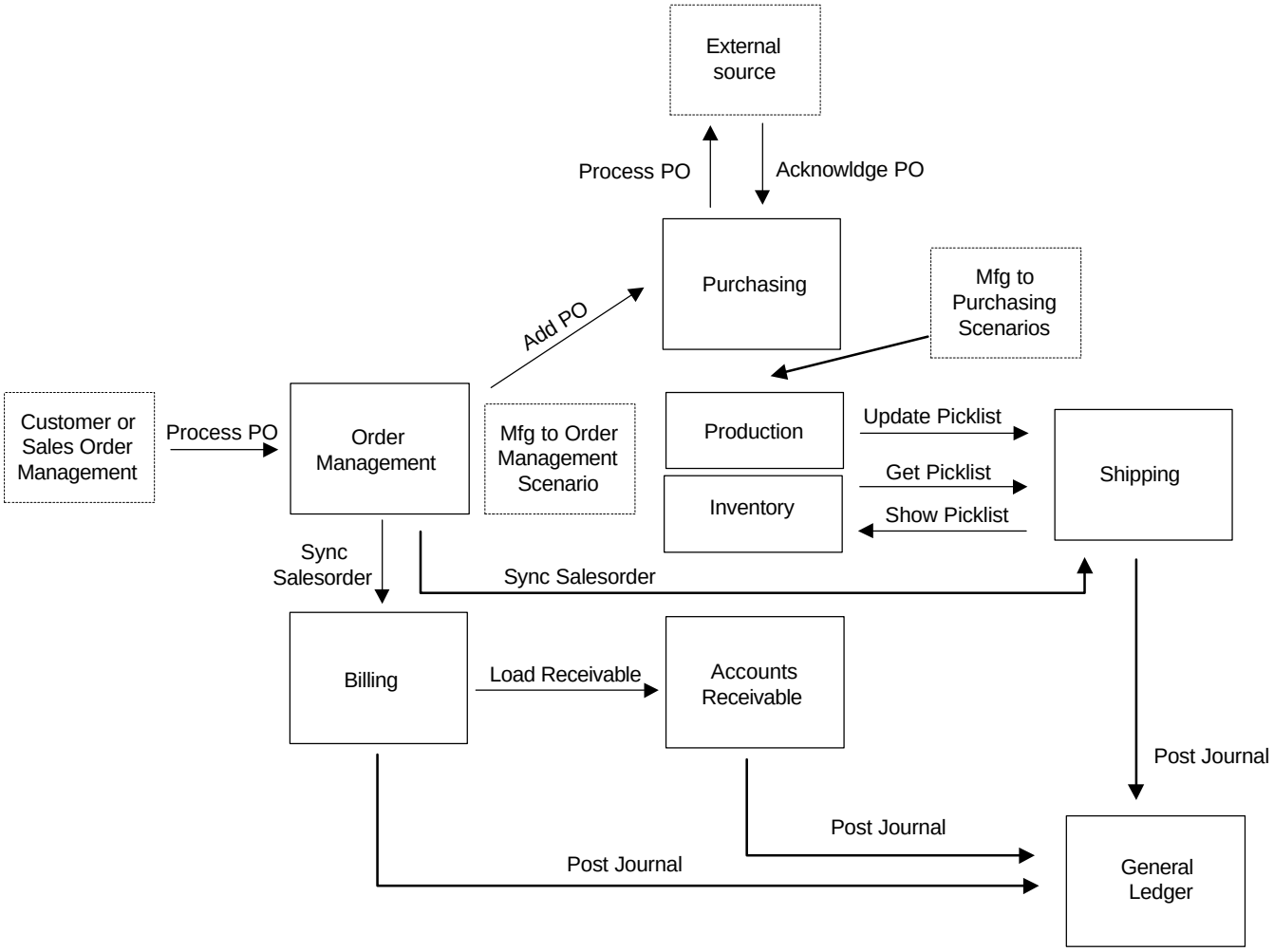
38.0 SALES FORCE AUTOMATION TO ORDER MANAGEMENT, INQUIRING ON ORDERS IN ORDER MANAGEMENT



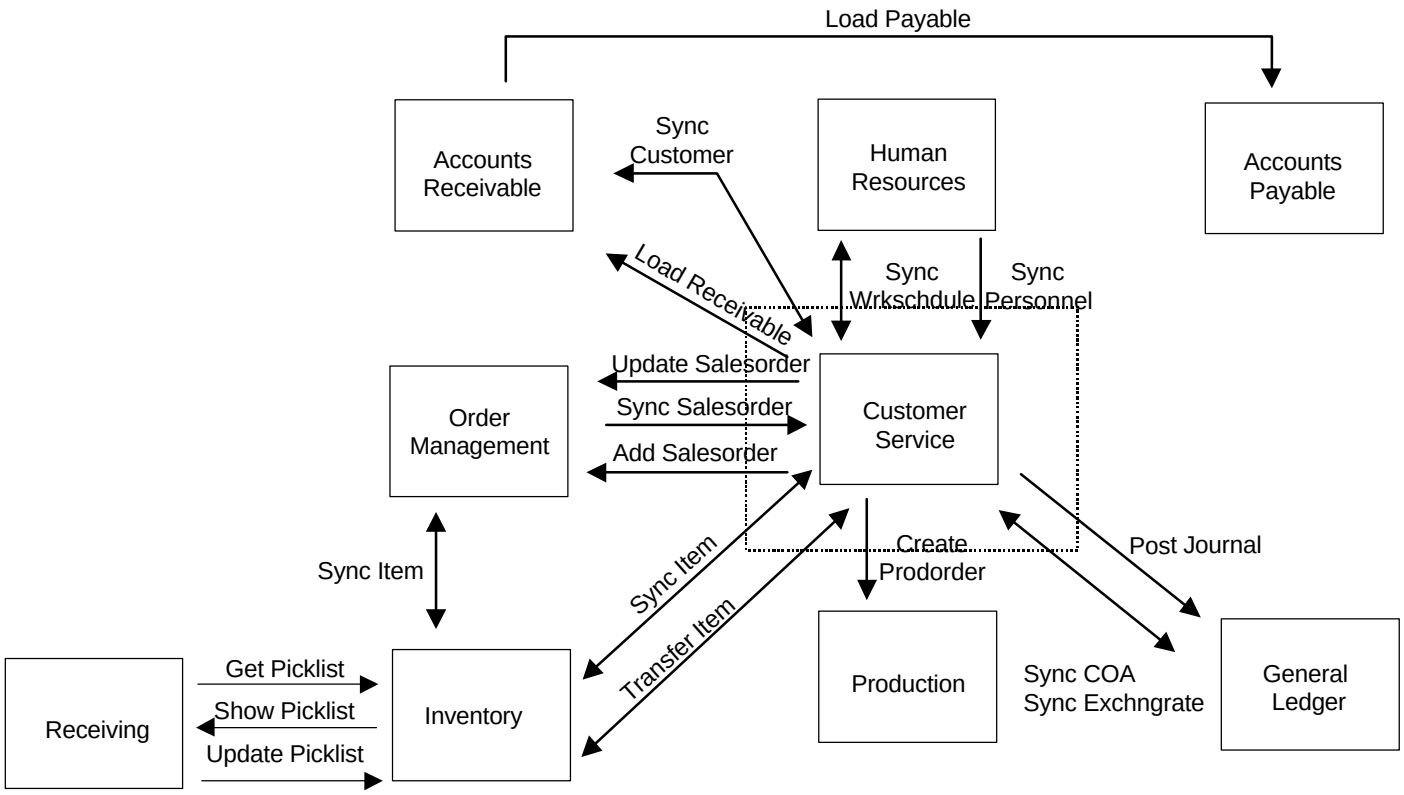
39.0 SALES FORCE AUTOMATION TO ORDER MANAGEMENT AND SHIPPING



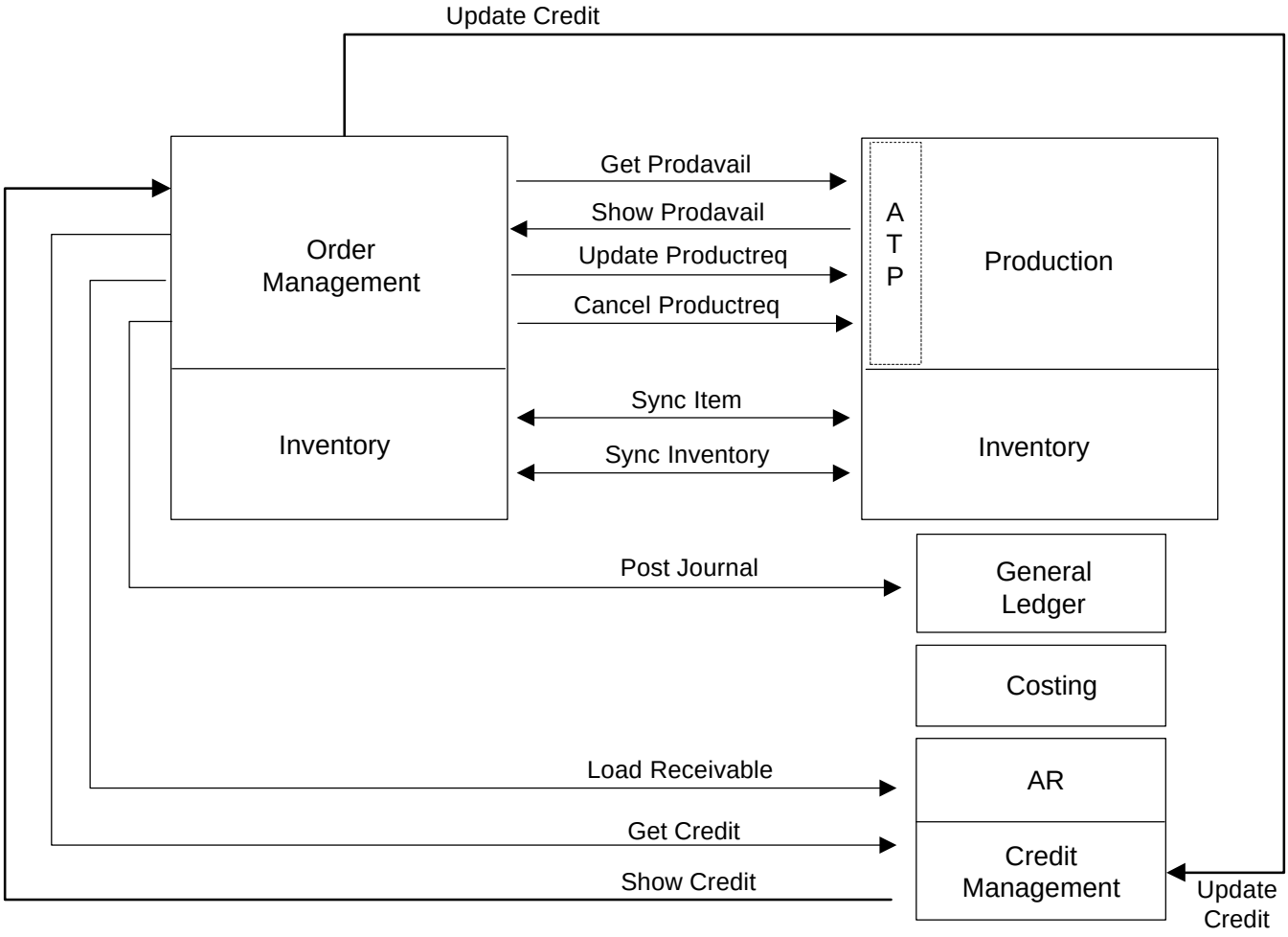
40.0 SUPPLY CHAIN INTEGRATION, MANUFACTURING TO PURCHASING, ORDER MANAGEMENT, BILLING, SHIPPING, AND FINANCIALS



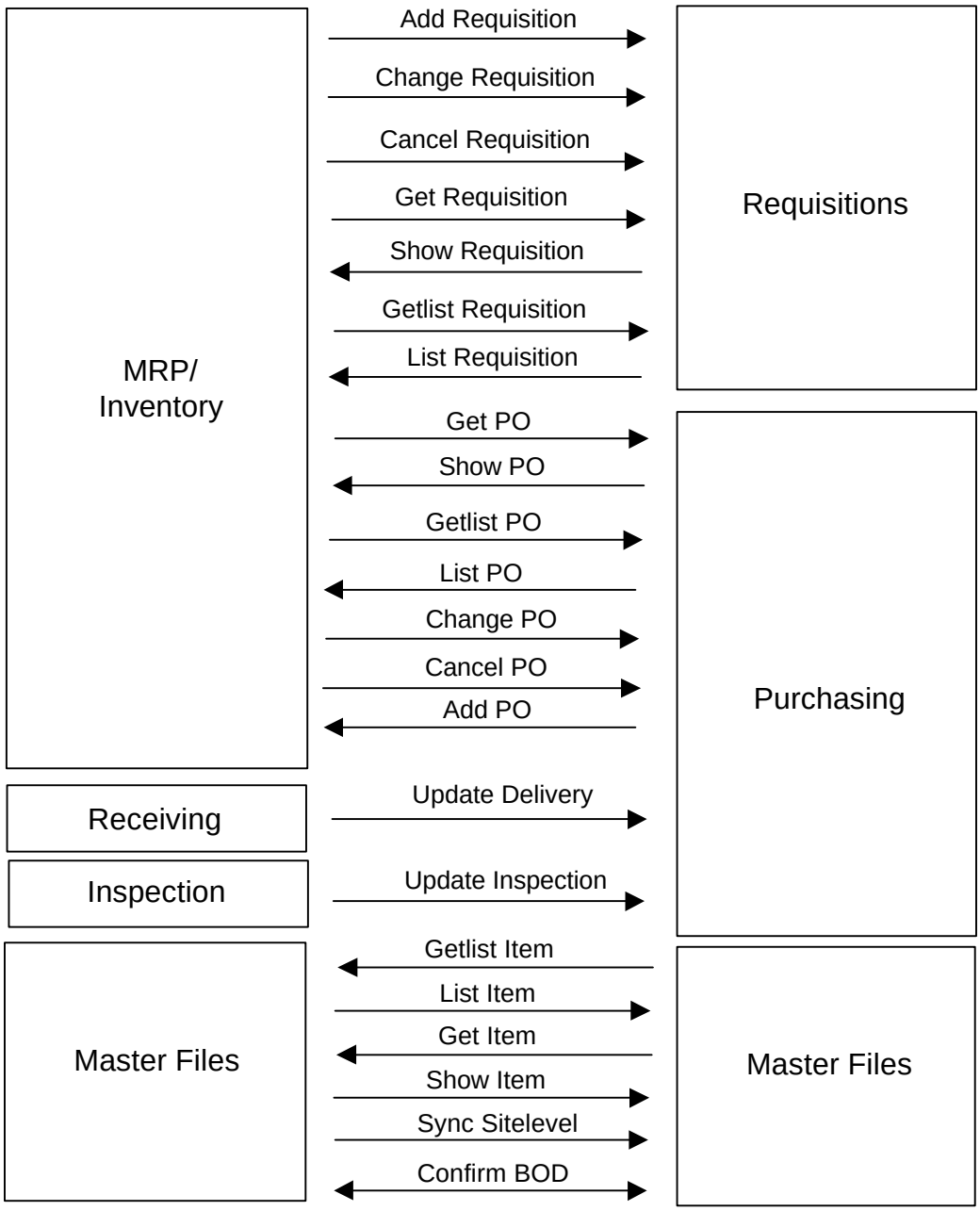
41.0 CUSTOMER SERVICE INTEGRATION, FIELD SERVICE, NO RETURNS



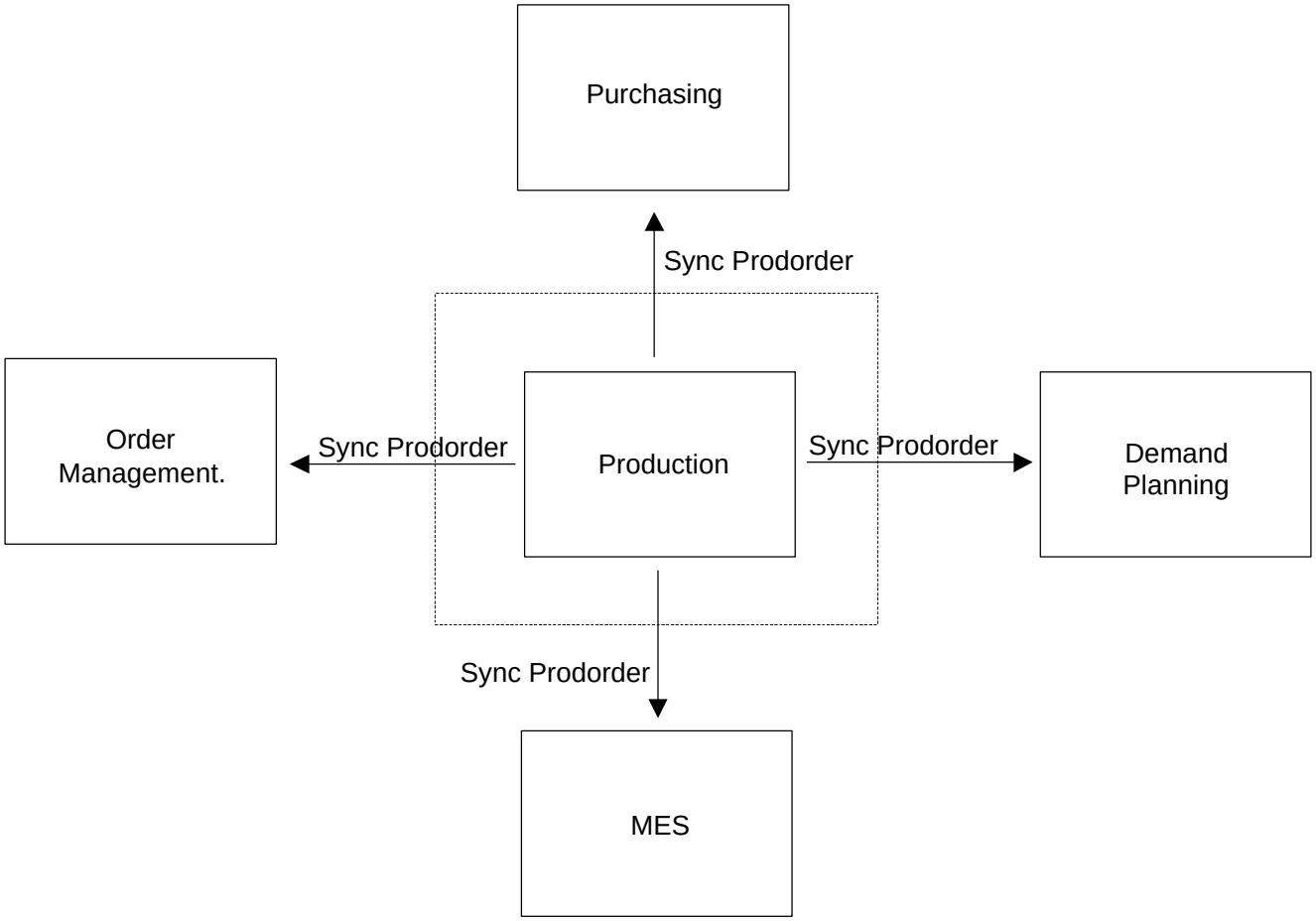
42.0 MANUFACTURING TO ORDER MANAGEMENT, FINANCIALS WITH MANUFACTURING, MAKE TO ORDER WITH CREDIT CHECKING



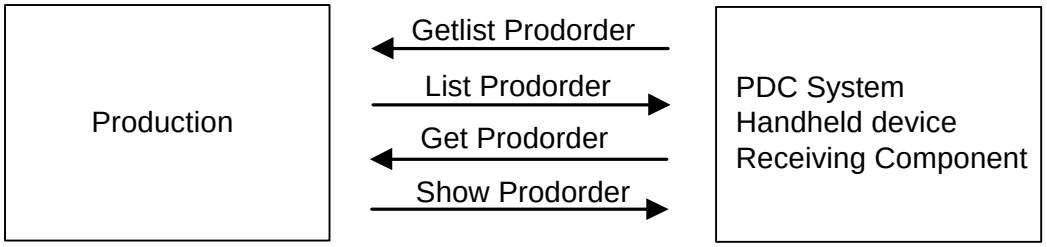
43.0 MANUFACTURING TO PURCHASING, RECEIVING AND INSPECTION IN MANUFACTURING, REQUEST/REPLY MODEL



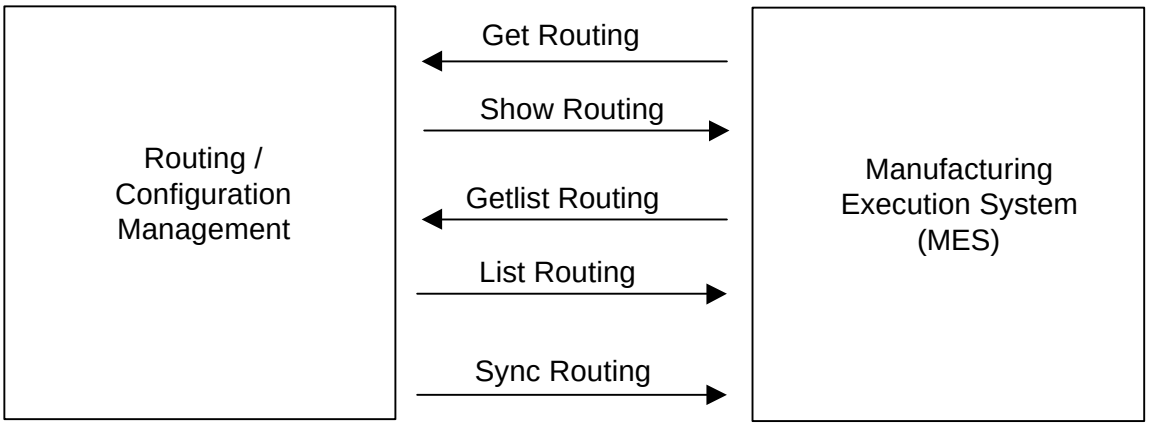
44.0 PRODUCTION SYNCHRONIZATION



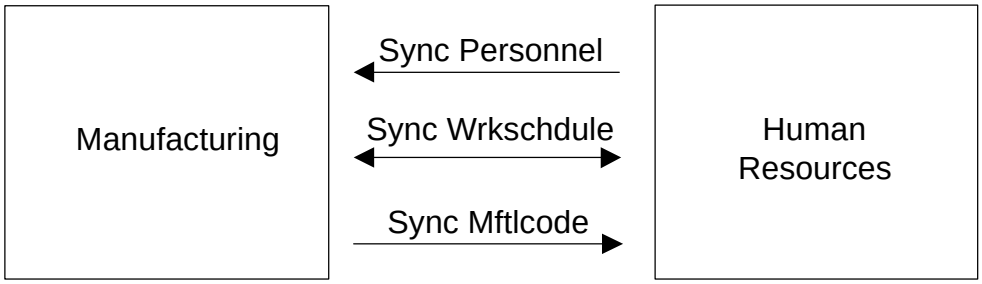
45.0 PURCHASE ORDER INTEGRATION



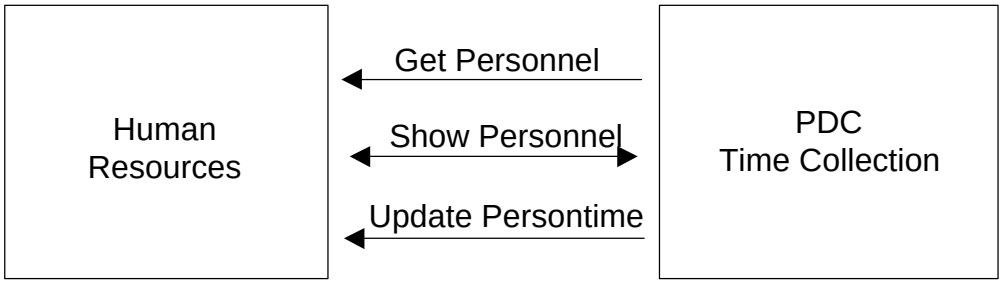
46.0 PRODUCTION ROUTING SYNCHRONIZATION



47.0 HUMAN RESOURCES INTEGRATION



48.0 HR TO TIME DATA COLLECTION



49.0 ENGINEERING CHANGES SCENARIO DESCRIPTION

49.0 Overview

This chapter describes the integration scenario for managing engineering information flows. Such flows may originate from many departments within an organization, including, but not limited to:

1. Contracts Administration -For new customer contract requirements for and Engineer to Order contract

2. Compliance -For new regulations for product safety

3. Manufacturing - For difficulty in mass production of a prototype that work in limited production.

4. Logistics - For items that are difficult and expensive to transport and handle.

5. Field Service - For items that fail in the in the field or have high warranty costs.

6. Marketing - For new customer requested features.

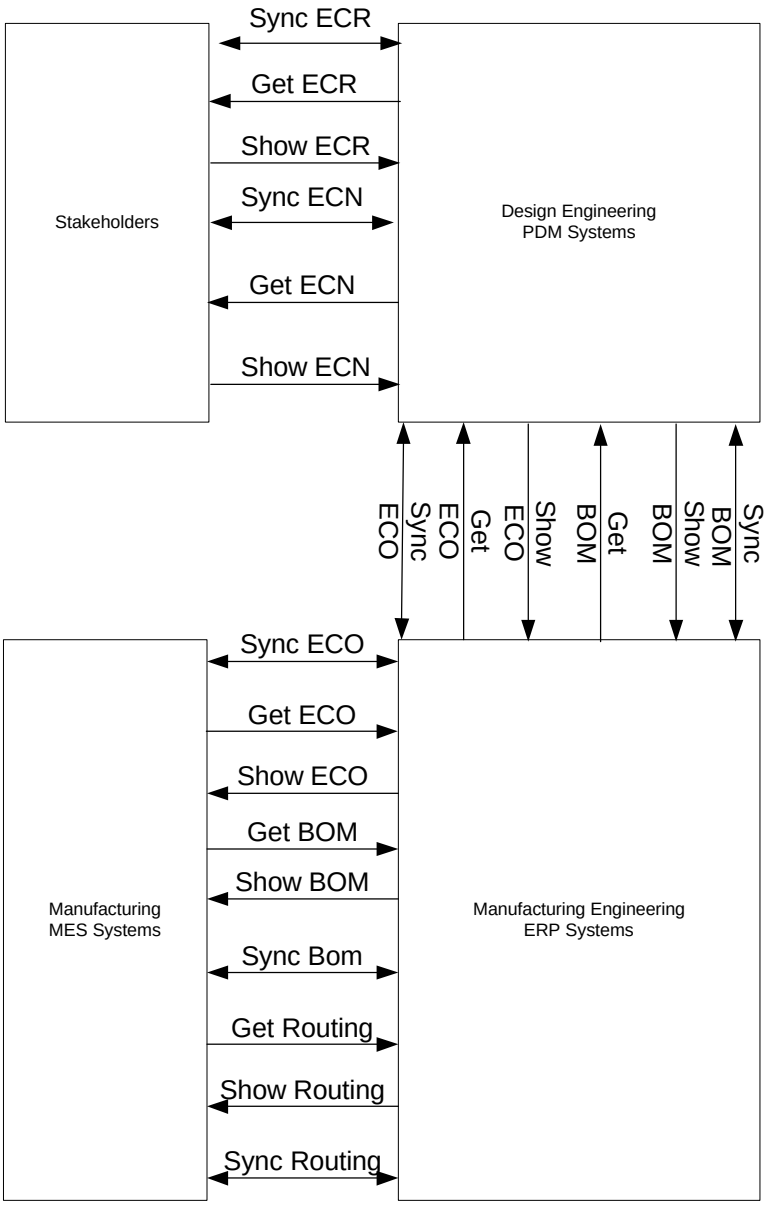
This is not a complete list but is meant to be a representative sample of activities that generate a engineering changes.

The purpose of this scenario is to enable the visualization of the components and the dialogs between components for this specific integration domain. This scenario is *not* meant to be the only correct model for integrating Product Data Management components to a other business software components. This is one model that may be used to design one's own integration based on the specific needs of an organization or group of organizations.

49.1 Scenario Diagram

The scenario diagram below contains the components involved in the interaction, the dialog flows or conversation between the components, certain assumptions about the sequence of events, and assumptions about the technical approach, for example, publish and subscribe. The next three sections will give a brief overview of the components, the sequence of events, and the environment.

Again, this is a MODEL to be used as a reusable design document, not a required approach.



49.2 Assumptions

This scenario assumes a loosely coupled, asynchronous approach to the integration. Transaction management is implicitly required but not explicitly defined.

This scenario describes a model for one or more Design Engineering components integrating with other common Manufacturing and Distribution components. The environment for this integration may be within a single enterprise, or across enterprises. There may be instances where the information in an engineering change request can be at the level of requirements, and desired behavior. As the engineering change moves through it's lifecycle, the information must become more and more detailed. As the information moves into the Engineering change order phase, it must contain all the information needed to manufacture the item or items

49.3 Component Definitions

This scenario contains three major components. Design Engineering, Manufacturing Requirements Planning and Manufacturing Execution. The definitions of these components are left to the designer but are assumed to contain the functionality as defined by what is commonly sold in the commercial market place today.

This heuristic definition is broadly accepted by the designers of the integration scenario and is a direct decision not to define how the processing takes place within any individual component. The component must be able perform the services defined by the business object document, but the internals of the component are not required or desired to be exposed.

The most important factors in defining these components is to ensure that the designer can communicate the requirements and design precisely enough to design the interfaces needed and their interrelationships.

49.4 Business Workflow (Sequence)

This scenario has two major events in the workflow sequence.

1) The First Business Event is the request for an Engineering Change that is made from any client department.

The synchronization process can ensure that all of the components in the integration scenario will have the same data necessary to communicate.

2) The second Business Event is the Creation of the Engineering Change request in the Engineering systems.

3) The Third Business event may be that the engineering change is communicated broadly to a set of contributors to a design.

4) The fourth business event may be that the engineering change is believed by the designers to meet the requirements stated in the specification. A notification is sent out to a list of approvers to sign off on the change before it can go to the next stage.

5) The fifth business event is that the approvers send in their approvals and the design change can be sent to manufacturing engineering to make any modifications to the assembly operations for any effected parts, or equipment.

6) The sixth business event is that manufacturing engineering are informed that the authorized method of manufacture has changed.

49.5 Exception Handling

This section is not intended to fully define the only acceptable methods for exception handling for this scenario. It is intended to be used as a guide by the designer to understand the intent and think through the issues the creators of this model had in mind.

The CONFIRM BOD shown in the scenario is the most obvious method for providing an application level feedback mechanism between business software components. This CONFIRM BOD is described in detail in section three, chapter two of OAGIS, however the specific use of the CONFIRM BOD may vary significantly from scenario to scenario.

The CONFIRM BOD in this scenario is intended to be used by the general ledger component to communicate to the sub ledger that the information it sent ***was received and understood***. If the information was not received or not understood, or contained errors of any type, the financial tracking for the organization may be incomplete or incorrect.

It is strongly recommended that the CONFIRM BOD is used in this scenario to prevent this potential problem.

50.0 ERP TO FINITE SCHEDULING AND MANUFACTURING EXECUTION SCENARIO DESCRIPTION

50.0 Overview

This chapter describes the integration scenario in order to enable the execution of Manufacturing Systems business software components to integrate manufacturing systems.

The purpose of this scenario is to enable the visualization of the components and the dialogs between components for this specific integration domain. This scenario is *not* meant to be the only correct model for integrating sub-ledger components to a General Ledger business software component. This is one model that may be used to design one's own integration based on the specific needs of an organization or group of organizations.

Several applications can provide a Dispatch List to the manufacturing system. The scenario diagram below shows an integration that involves Finite Scheduling as the source of the Dispatch List.

Possible sources for a Dispatch List may be:

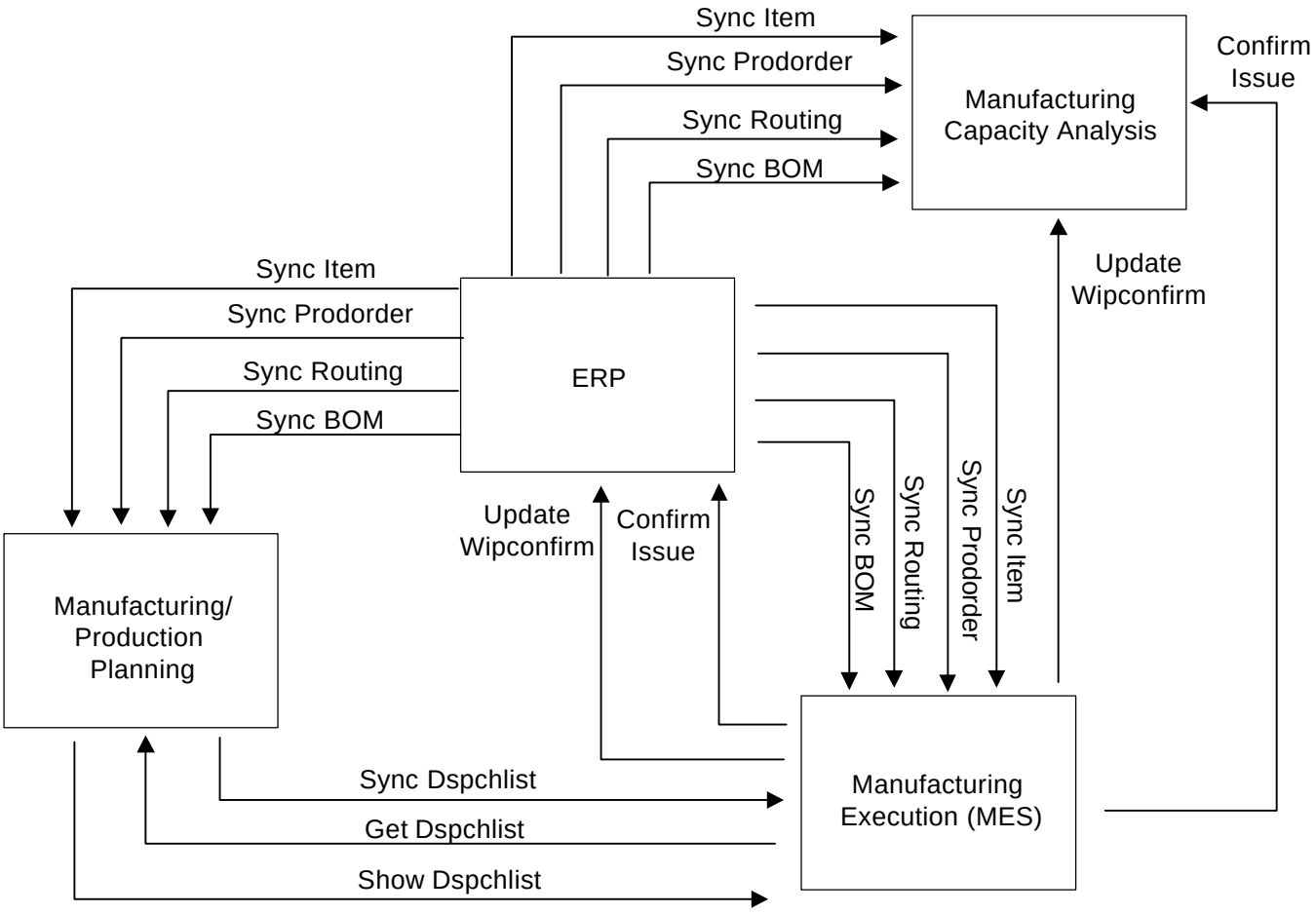
1. Finite Scheduling
2. ERP

This is not a complete list but is meant to be a representative sample of activities that generate a journal entry.

50.1 Scenario Diagram

The scenario diagram below contains the components involved in the interaction, the dialog flows or conversation between the components, certain assumptions about the sequence of events, and assumptions about the technical approach, for example, publish and subscribe. The next three sections will give a brief overview of the components, the sequence of events, and the environment.

Again, this is a MODEL to be used as a reusable design document, not a required approach.



50.2 Assumptions

This scenario assumes a loosely coupled, asynchronous approach to the integration. Transaction management is implicitly required but not explicitly defined.

This scenario describes a model for production planning.

The environment for this integration is typically within an enterprise and within a division.

This scenario also assumes that one application will maintain the master data for integration. The Business Object Documents in the scenario diagram are described in section three of OAGIS.

50.3 Component Definitions

This scenario contains four components.

1. The production planning component
2. The manufacturing component,
3. The manufacturing execution component, and
4. The ERP component.

The definitions of these components are left to the designer but are assumed to contain the functionality as defined by what is commonly sold in the commercial market place today.

This heuristic definition is broadly accepted by the designers of the integration scenario and is a direct decision not to define how the processing takes place within any individual component. The component must be able perform the services defined by the business object document, but the internals of the component are not required or desired to be exposed.

The most important factors in defining these components is to ensure that the designer can communicate the requirements and design precisely enough to design the interfaces needed and their interrelationships.

50.4 Business Workflow (Sequence)

This scenario contains the following events in the workflow sequence.

1) The first business event is an order is received from an external customer. When a new Item is added to the ERP system, it is automatically synchronized out to the other applications.

2) The next step is to create a production order from the sales order from the customer.

3) These Production orders are then sent to Capacity Analysis component, and Scheduling component, this includes the Production order, Bill of Materials (BOM) and Routing.

4) The Scheduling component creates a list of tasks to be done, a Dispatch List, by the manufacturing Execution system in order to fulfill production orders.

5) If a disruption occurs in the manufacturing process a Manufacturing system can request to Get the Dispatch List and receive a Show of the Dispatch List.

6) Upon completion of these tasks, the Manufacturing system updates the ERP and Capacity Analysis systems are updated with the WIP in progress and the confirmation of issues of materials.

50.5 Exception Handling

This section is not intended to fully define the only acceptable methods for exception handling for this scenario. It is intended to be used as a guide by the designer to understand the intent and think through the issues the creators of this model had in mind.

The CONFIRM BOD shown in the scenario is the most obvious method for providing an application level feedback mechanism between business software components. This CONFIRM BOD is described in detail in section three, chapter two of OAGIS, however the specific use of the CONFIRM BOD may vary significantly from scenario to scenario.

The CONFIRM BOD in this scenario is intended to be used by the components to communicate the information it sent ***was received and understood***. If the information was not received or not understood, or contained errors of any type, the financial tracking for the organization may be incomplete or incorrect.

It is strongly recommended that the CONFIRM BOD is used in this scenario to prevent this potential problem.

51.0 COMPUTERIZED MAINTENANCE MANAGEMENT SYSTEM (CMMS) TO FIELD DEVICES