2 Scope of Workflows

Abstract *Chapter 2 is about the scope of workflow automation in the printing industry. In many cases, the term workflow refers only to the work steps within production, i.e. in prepress, press and postpress. In this chapter, however, the term is defined more broadly, which is why areas such as the print buyer's purchase order, materials management, or shipping the finished products to the customer are included.*

Keywords *Management Information System, Print Job Submission, Web-to-Print, Procurement, Workflow Management System, Printing Technology.*

From the print provider's perspective, there are workflows with

a) internal interfaces/processes which are organized and executed inside the print shop
b) interfaces/processes that relate to external partners

> **Definition**
> An **interface** is the point where two processes or software components interact.

Figure 2.1 shows the most important interfaces of a print provider.

The interfaces b) must always connect to internal workflows. That is, workflows can have internal interfaces and processes only, or they can have external ones as well. However, this distinction depends on the range of the considered workflow. For example,

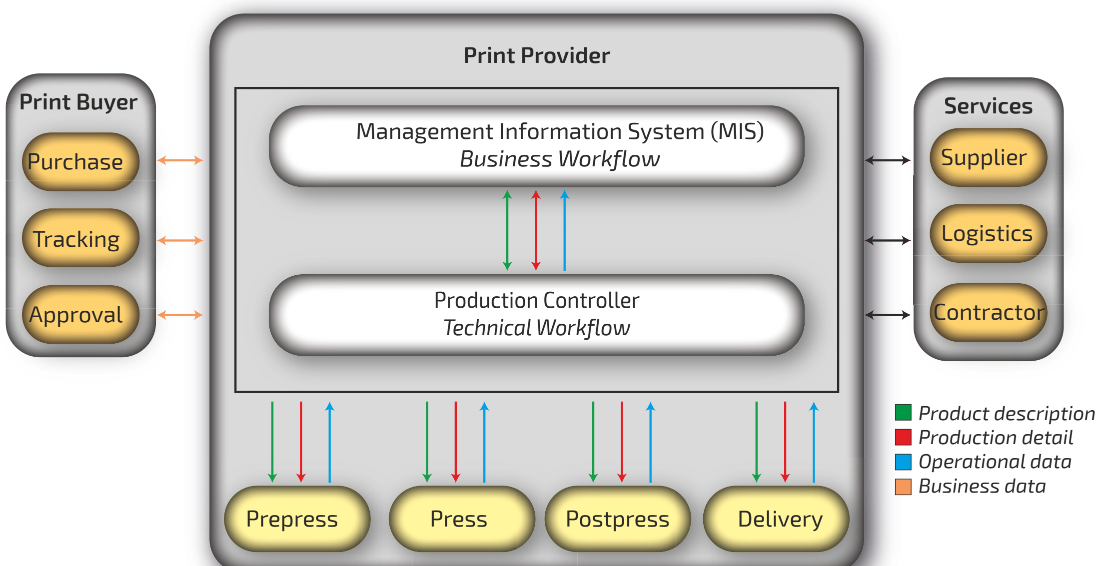

Figure 2.1: Interfaces of a print provider to external business partners and internal production departments

T. Hoffmann-Walbeck, *Workflow Automation*, SpringerBriefs in Applied Sciences and Technology, https://doi.org/10.1007/978-3-030-84782-1_2

> **Definition**
>
> A **workflow** denotes the sequence of production and business processes, during which documents, information and task descriptions are passed from one participant to another for action according to a set of procedural rules.

if you look the *plate making* workflow, there are only internal processes at first glance (like imposition, RIPing, imaging), provided it is not done by some external subcontractor. An approval process of the imposed sheets with the customer certainly would count as an external interface. The same holds with a notification that the platesetter has imaged the plates. Thus, it is a matter of defining the range of the workflow if it is internal or external. In the end, if you consider the entire print production as one huge workflow, it always has at least two external interfaces: *purchase* order and *delivery.*

Having said this, I still distinguish between the two. In Section 2.1, let's start with the business workflow in a print shop controlled by a *management information system (MIS)*. I am putting this topic at the beginning because the MIS is often the central communication hub between internal and external workflows. Section 2.2 deals with the communication between the print buyer and the print provider. Next, let's discuss the print shop's material procurement (2.3). Section 2.4 is about the internal print production workflows, which are currently the most dominant area of workflow automation. This is where it all began. As a sub-topic, I will present *workflow management systems* (WMS), i.e., controllers that manage one or several production processes (partly) automatically under a common user interface. Section 2.5 is about the delivery of the finished products. Finally, the chapter ends with other interfaces to external partners to whom a print provider has outsourced parts of its business and production.

Some people claim that the external interfaces allow for major increases in efficiency in the future, while the internal production processes are already largely automated. This is true in theory but does not really apply in practice.

2.1 Management Information Systems (MIS)

A *management information system* is a general term for computer-aided systems that provide information about commercial business processes. An *enterprise resource planning system (ERP)* goes one-step further: the software also controls production processes while making the best possible use of resources. This is sometimes called a *manufacturing execution system (MES)*. Since all three terms are often used interchangeably in the graphic arts industry, I will not differentiate between them.

The MIS/ERP/MES is the topmost workflow controller in the print shop. It is also sometimes referred to as an *agent.* An *agent* is a workflow component that initiates the workflow communication by writing the initial workflow data (i.e., the job ticket, see below) for a print job.

First, I will look at some key MIS features, followed by the interfaces of an MIS (2.1.2). However, these topics are not disjointed but overlap to some extent.

2.1.1 Main Features of an MIS/ERP

The main tasks of an MIS are:

- Price estimation and job management
- Production planning, controlling and scheduling
- Reporting, controlling and visualizing the business status for management decisions

When you launch an MIS, you normally see a list of all current print jobs managed by the MIS. The order management operator can sort this list in different ways. A core function of the job management system is the creation of new print jobs of different types and estimating their prices. For the latter, an MIS must be able to compile the technical processes of the production for each job. The process planning for the production can go quite deep. For example, the MIS might suggest imposition schemes, even for gang forms. Moreover, the delivery date will be registered.

Traditionally, after the MIS order manager has generated a job and the print buyer had agreed on the price, a paper-based *job ticket* is printed, which in turn is then forwarded to the prepress department. The job ticket contains the product description as well as production instructions (not only for prepress). A paper-based job ticket is not state-of-the-art anymore. Nowadays, the MIS sends the job data including the basic production parameters digitally to a prepress *workflow management system* and to other production systems. The *Job Definition Format* (JDF) is widely used for this purpose (see Section 4.3). The digital transfer of job tickets is more efficient and less error-prone. In particular, data does not need to be entered twice anymore.

Usually it is also desired that an MIS displays the production status of all jobs. In order to achieve this, operational data from the production floor must be sent back to the MIS (see Section 2.4.3). This also allows the print provider to control the production costs per job and compare them with the estimated price set in advance. Thus, the profitability can be calculated for each job or job type (post-calculation). Furthermore, it enables production control, job tracking, and the automatic generation of meaningful management reports on things like turnover development, delivery performance, machine utilization, waste analysis, or more abstract values such as *overall equipment effectiveness* (OEE).

Most MIS have optional modules that can be licensed separately, such as a web shop system, an electronic planning board, or warehouse management software. Since these three tasks can be

Definition

In a **gang form** (parts of) several print jobs possibly from different customers are placed on the same sheet.

Definition

A **job ticket** is a document that details the specifics of an order. It can contain a description of the intended print products as well as instructions for their production.

Definition

A **workflow management system (WMS)** is a software system that defines, creates and manages the execution of workflows. It also controls one or more devices which are able to execute processes.

Note

The **overall equipment effectiveness (OEE)** measures the manufacturing performance relative to its full potential, i.e., the manufacturing productivity.

performed with external modules from a different vendor (and be integrated with the MIS), I discuss them not in this MIS section, but rather in Sections 2.2.2, 2.4.4 and 2.4.5, respectively.

2.1.2 Overview of Interfaces

Figure 2.2 shows the main interfaces of an MIS except for the production interface. Note that many of these might not be part of each MIS configuration in a printing plant. They can be either optional modules of an MIS or are not included in the MIS at all. The modules are increasingly merged (e.g., print buyer, web-to-print, and customer relationship management).

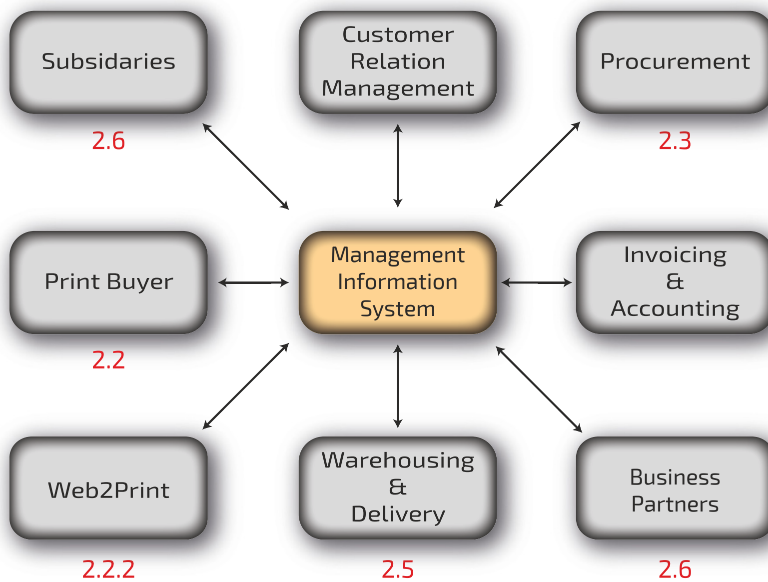

Figure 2.2: (External) interfaces of an MIS

In Figure 2.2, the section numbers are marked for those interfaces that I discuss in the following sections (2.2 through 2.6).

The top arrow between the print buyer and the print provider's MIS in Figure 2.1 represents all job-related communication issues such as quotes, purchase orders or invoicing (see Section 2.2.1). It does not include marketing activities, which are covered under the term *customer relationship management* (CRM). Note that CRM is often understood as the sum of all information and activities around the customer, including order-related correspondence.

The most dominant external communication of an MIS is the interface to the internal production (see Figure 1.1 and Section 2.4). Much data which has been already registered in the MIS must be passed on to the production. Examples include:

- Product specifications such as product type, printing substrate, product parts, binding, colors, number of pages, dimensions of pages, finishing processes (e.g., laminating or varnishing), etc.
- Data about necessary production processes such as printing technology, process details, machines to be used, print runs, etc.
- Organizational issues such as customer information, production schedules, and deliveries.

> **Note**
> The **product specification** is a description of how the print buyer wants the product to be in the end. It is independent of the necessary production processes.

> **Note**
> A **product part** is a component of a product. For example, a softcover book consists of two parts: cover and content..

The description of the processes can be quite general or highly detailed. For example, not only the process names “sheet layout preparation” and “imposition” can be transferred to prepress, but also imposition schemes, precise folding positions, or even ready sheet layouts. The depth of this data exchange is important for evaluating the integration between MIS and production system.

The interface between MIS and production has been subject to lots of debate since the beginning of the 21st century. By now, it is a well-established communication channel. The paper-based job tickets that were common in the past have been more and more replaced by electronic job tickets. A smooth MIS-production interface is therefore essential for workflow automation. The data formats between these two instances have also been standardized, primarily in the form of JDF and XJDF (see Sections 4.3 and 4.4).

Note that the data transfer between the MIS and the production controller and devices is not a one-time matter per print job, i.e. when the MIS hands over the electronic job ticket to the production. Often, the print buyer changes the order afterwards, which requires updating. Sometimes, the order manager can even control some devices directly, like reprinting plates, for example. Moreover, the interface normally is bi-directional. The devices and production controller will a) send back operational data to the MIS (see 2.4.3) and b) send data for updating the job ticket information.

An update might be necessary when some production operator changes some values in the job, or some controller/device adds some data automatically. Normally, the MIS job ticket does not cover all parameters for the production, and those are filled in later with default values, for example.

> **Note**
> The bidirectional electronic interface between the MIS and production systems is standard nowadays. The MIS sends product/production data to the production, and the production sends status information back to the MIS.

Please keep in mind that these standardized data formats may be used only for communication between the MIS and the production systems – internally, the data structures may be completely different. In fact, much information will be stored in private databases of the MIS and of the production system, but this architecture still allows the user to connect MIS and production systems from different vendors.

2.1.3 Current Trends in MIS/ERP

In recent years, many new features have been added to management information systems. I want to highlight just two of them.

In the past, MIS server software and the associated client software were installed on the company's own computers (on premises). In recent years, however, more and more systems have become cloud-based, and clients can use their local browsers to communicate with the remote system. Initially, there were many concerns about data security, but these have diminished.

The cloud can be either public or private, i.e., based on the public internet or on some private intranet. Private clouds are relevant mainly for big companies that strive to interconnect several subsidiaries. Multi-site production and internationalization presumes MIS software that can handle not only different languages and currencies, but that most notably can organize production across several locations with the lowest cost. To accomplish this, some very large companies don't install an off-the-shelf solution from some MIS vendor, but rather build their own system instead. Still, they often use some functionality of a purchased MIS product. That is, they build their own MIS by deploying an API (application programming interface) from some commercially available MIS that they bought from an external vendor.

Another important trend concerning MIS is to allow print buyers to input data into the software directly. Thus, the print provider no longer needs to enter all the data to specify the requested product. This capability makes the following features necessary:

- The MIS can grant the customer access in a secure manner.
- The MIS can estimate production costs fully automatically.
- The print buyer is willing and able to enter the product data into the system.

> Definition
>
> **Electronic Data Interchange (EDI)** is a universal term for the electronic transfer of information from one computer system to another.

This procedure surely is only valid for long-term customers of a print provider. Alternatively, *electronic data interchange* (EDI) between the print buyer's ERP and the print provider's MIS, based on a master agreement between the two parties, can be even more effective (see 2.2.1).

2.2 Print Buyer - Print Provider

The information exchange between the print buyer (PB) and print provider (PP) is of a diverse nature. I like to classify them roughly according to the type of print job, such as:

- Unique print jobs
- Template-based print jobs

The first bullet point means that the customer requires a unique print product, not only in terms of content but also concerning the product features like dimension, inks, printing substrate, embellishments, and the like. For that, a special quote must be prepared. Let's discuss this MIS-based approach briefly in Section 2.2.1. Template-based, on the other hand, means that the degree of freedom concerning the artwork and the product might be limited. This case will be the topic of Section 2.2.2. Both kinds might include proofing, portals, artwork submission and notifications, which are covered in Section 2.2.3.

Note

In this text, **artwork**, **print data** and **content data** are used synonymously.

2.2.1 Business Transactions

Figure 2.3 shows some typical data exchanges between print buyer and print provider. Each of these *business objects* or *business data* between the two business partners can be defined by a *PrintTalk* element (see Section 4.4). This protocol can be the basis of an integration between PB and PP. For both the PB and the PP, however, the more critical point will be the extent to which the automation of these business processes can reduce the number of human touch points.

Figure 2.3: Common data interchange between print buyer and print provider

Traditionally, these business objects have been informal and unstructured. The communication channels are telephone, e-mail, postal mail, or even telefax. Direct sales is another high-touch method of placing orders. For the artwork, e-mail attachments or courier/mail services for sending physical media such as CD/DVD or USB sticks are common (Figure 2.4 - ❶). The communication is carried out between PB and PP employees. On the PP side, staff normally re-keys the customer's data into an MIS. These communication channels are not particularly effective. Phone calls and e-mails can lead to misunderstandings and subsequent inquiries. Their potential for automation is low. A change of communication channels should be considered.

To send artwork via an e-mail attachment, files must not be very large. Another option, which has also been used for many years, is to transfer data using the FTP protocol (Figure 2.4 - ❷). For this, the PP must set up an FTP server and the PB can then transfer data to the server using FTP client software.

Note

Business objects or **business data** describe business transactions between print buyer and print provider, such as a request for a quote, a purchase order, or an invoice.

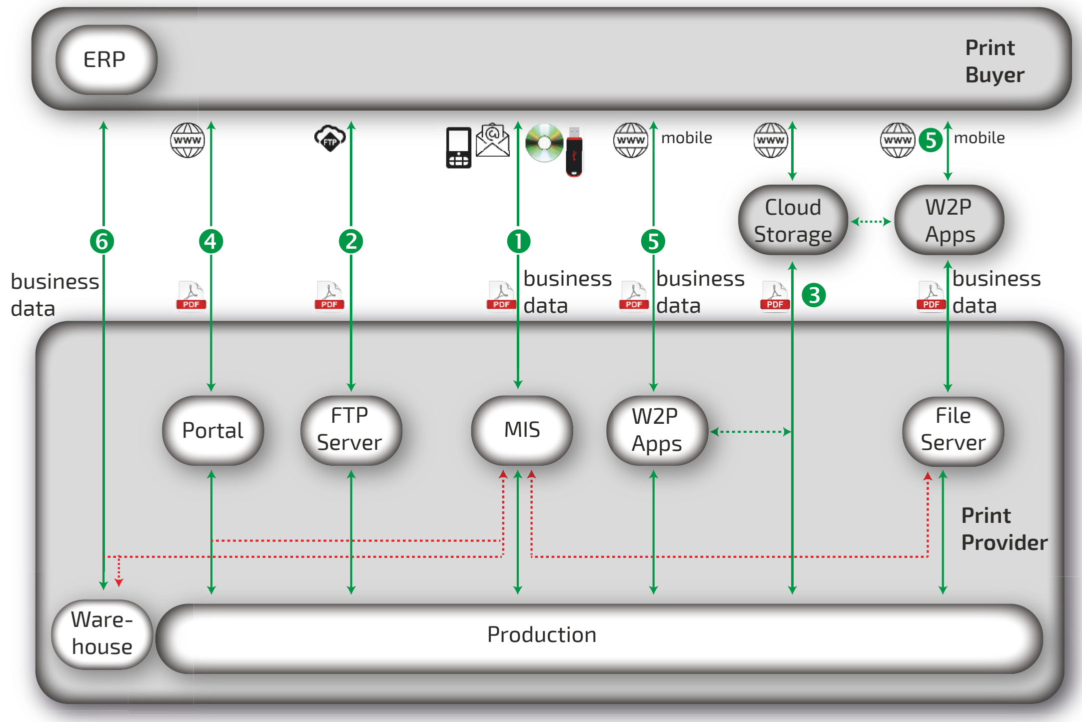

Figure 2.4: Job submission between print buyer and print provider

In recent years, as we all know, the transfer of data via a file hosting services has become increasingly popular. Third-party providers make storage space available for this, which the PB and PP can then use jointly. It is also referred to as *cloud storage* (Figure 2.4 - ❸).

Content data submission via Web-to-Print (W2P) and portals is much more effective. The main advantage is that the uploaded content can be routed to the job data without human interaction. Manual assignments between job data and artwork are no longer necessary. They are not only time-consuming, but also a potential source of errors. In addition, the transfer of the content data is documented and equally transparent for the parties involved. This is particularly important for revisions. Finally, the artwork file can be preflighted automatically, and properties can be checked according the job data (e.g., number of pages or page sizes). The PB will then get immediate notification concerning possible serious errors. Even further processes such as trapping and color management can be automated. In Figure 2.4, arrow ❺ represents this communication path between PB and PP (see next section for more details).

In ❹, the PB uploads an artwork file with a web browser using HTML 5 and some associated programming languages to the portal server. PrintTalk (see Section 4.5) can be used as an underlying protocol between portal/W2P and the PP's MIS. Inside PrintTalk's

ContentDelivery business element, a URL can be defined where the data can be found. The portal/W2P system will send this element to the PP's MIS, which can then respond with a *ContentDeliveryResponse* PrintTalk element.

Some business transactions like *request for quote, quote, purchase order* and o*rder confirmation* are usually carried out before the PB submits the artwork to the PP. With W2P, both the business data and the artwork are sent simultaneously. There is, however, a third option: The print buyer (or some creative agency) submits artwork to the PP and thereby triggers an order implicitly. Usually, a prepress operator or a prepress workflow management system creates the order, and software helps to describe the print job by analyzing the file. For example, the number of pages, page size, colors can be extracted from the PDF (see light green arrow in Figure 2.11). There is still important job data missing, such as the number of requested copies, the printing substrate, or the binding type. The prepress operator or the MIS order manager must get this information from the PB in retrospect. This procedure contradicts the theory but is common practice. It might become even more popular when more product description data is integrated into the PDF (see Section 4.7).

Finally, I want to mention the option of an *electronic data interface (EDI)* between the print buyer's enterprise resource planning (ERP) system and the print provider. This allows a PB to easily retrieve or order products directly via its internal ERP system. This interface is shown graphically in Figure 2.4 - ❻). Such an EDI interface is normally a special solution combining the PB's and PP's systems, i.e., it is customized. It might be connected to the PP's warehouse management software and MIS. However, this type of purchase order is only reasonable for major and long-term relationships between customers and print providers.

2.2.2 Web-to-Print (W2P) and Apps

In the last section, the traditional onboarding methods started with unstructured information from the PB that some MIS order management operator had to re-key. Moreover, the PB or its agency send artwork separately from the job data. Both data sets must then be linked by somebody at the PP, which increases labor costs.

> **Definition**
> W2P is a commercial printing platform for job and artwork submissions using web sites.

As is well known, Web-to-Print involves creating, ordering, producing and submitting print jobs via the internet. That is, it encompasses e-commerce aspects as well as the creation and submission of the artwork in most cases. Web-to-print automates the submission of print jobs and reduces the number of human touch points for the print provider (e.g. create quotations, process purchase orders).

The main characteristics of a W2P system are:

Definition

With **responsive web design**, the function, design and content of a web page might change in accordance with the screen size and resolution of the computer monitor, tablet, smartphone, etc. being used.

Definition

Business-to-customer (B2C) denotes a business interaction between an end consumer and a company, while **business-to-business (B2B)** refers to a business relationship between two (or more) companies.

Note

A print provider that gets orders (nearly) exclusively via one or several W2P shops is called an **online printer**.

- Online pricing, usually online billing (B2C)
- Integration with the customer's ERP (B2B)
- Online customization via pre-defined templates
- Uploading of print data (PDF, text, images)
- Online purchase orders
- Restrictions in product features
- Responsive web design (B2C)

W2P shops and appropriate mobile apps are mainly used for commercial printing, but also – to a lesser extent – for packaging printing, especially for label printing (web-to-pack). Niche markets are deploying this technology as well. The publication of photo books should be mentioned in this context.

In this section, I will go neither into the technical setups of a W2P shop nor into the business opportunities and risks of doing so. In addition, I will not cover the different features and requirements of public B2C and dedicated B2B systems. I will only discuss some fundamental data flow aspects.

One business condition, however, is so fundamental that is has a big impact on the PP's workflow: For online printers, the average value of a purchase order is usually quite small. On the other hand, there are many of these jobs to be processed each day, especially in to B2C environments. With B2B, repeat jobs are common. Both situations imply that order-related activities should be automated as much as possible by minimizing the number of human touchpoints per order. Thus, all the workflow automation options described in this book are particularly important for online printers. Actually, online printers are often pioneers in workflow automation and in some cases the initiators of these new, more efficient production processes. Without workflow automation, W2P is simply not profitable.

Definition

The file format **CSV** stands for **Comma-Separated Values** and describes the structure of a text file for storing or exchanging spreadsheet data.

Definition

XML stands for **Extensible Markup Language**. It is a very popular way to structure any data in text format. Standard data formats like JDF and XJDF are XML based. Vendor specific XML is also called "private XML".

The W2P system will inform the PP's MIS and/or workflow management system (see section 2.4.2) about incoming orders. Different data formats may be used, such as PrintTalk with JDF or XJDF, CSV, or private XML (see Chapter 4) to store the desired job data. XJDF might have some advantages over JDF, in particular because a XJDF file can store information about several products, while a JDF file always contains information about a single product only.

To exchange these kinds of data, the W2P server is run by the PP on-site or by some external service provider. If a PP operates its own W2P store, the W2P software is often an optional module of the MIS or prepress production system. In these cases, the interfaces between the modules are obviously internal. For example, the systems could simply access a common database. Of course, there are also PPs that install MIS, production system(s) and W2P software from different vendors.

It is self-evident that MIS, W2P and the production systems must be able to communicate with each other. From the system integration point of view, it is desirable that the vendors deploy standard data formats for this purpose (such as JDF, XJDF, or PrintTalk). This provides the highest level of flexibility in case one of the modules should subsequently be replaced without too much effort. However, even if some of the modules allow only vendor-specific XML, this does not mean that all is lost since XML documents can be easily transformed with help of an XSL transformer. For example, one could convert a private XML document to some standard JDF document or vice versa, assuming that the private XML document contains appropriate information for the JDF.

> Definition
>
> **XSL** stands for **Extensible Stylesheet Language**. It is used, among other things, to translate/ transform an XML format into another XML or any text format.

A W2P system is not a monolithic block but usually consists of different modules that are connected to the system via API interfaces. Typically, these modules might be:

- Graphic engine for creating templates and filling customizable templates online
- Rendering engine for generating PDF documents and/or image files
- Preflight engine for checking and making uploaded files production-ready
- DAM for providing digital assets such as images

> Definition
>
> **DAM** stands for **Digital Asset Management**. It allows the storage, retrieving, organizing and manipulation of digital assets such as images.

In addition, of course, there are e-commerce functionalities such as pricing, handling shopping charts, managing previous orders or even a private environment for each customer, data analytics, and interfaces to online payment providers and possibly to shipping companies.

2.2.3 Notifications and Approvals

Processes have several different output states – in the simplest case “success” and “failure”. Often, some sort of threshold values are set to map the whole range of quality levels up to one of these two extremes.

Most of these validations concerning the processes are handled within the production environment. The checks are carried out either by some operator or some automatic quality assurance mechanism. For example, after imaging (and developing) a print plate for offset printing, either an operator will (occasionally) visually inspect a CtP control strip on the plate or control the tonal values with a hand-held measuring device (mobile plate scanner), or some special image analysis software checks the imaging quality of the printing plates automatically on a conveyor line.

The results of some processes are checked for some kinds of product, but not for others. Trapping in prepress is normally done fully automatically in commercial printing and not inspected by anybody, while in the packaging industry this process will be checked and corrected visually by an operator (because of the higher usage of spot colors and non-standard color separations).

Reviews of processes can also be formally expressed in job tickets such as JDF.

These examples should suffice for internal quality assurance within the production operation. Next, let's take a look at the situation where external people review and approve/reject some interim results – i.e., some resources – during production. The approval stakeholders are the PP, the PB, content providers, and production agencies. On a sidenote, as more and more people work remotely from home, the distinction between internal and external approvals is becoming less significant, at least as far as the underlying technology is concerned.

Traditionally, there are the proofs (i) to (v) of semi-finished products (see below). In this list, I am omitting the design, content and color validations during the creation of the artwork, even though the proofing process is extremely important in this area. The reason for this is that it usually happens independently of the print provider, namely between the print buyer, design agency and photographer (unless the print provider also runs the agency). Moreover, often the focus is more on collaboration. For that, a workspace in the cloud is created, which allows the parties to jointly edit and review online.

i. Checking the PB's artwork for technical production suitability (preflight)
ii. Verification of the print-ready artwork before imposition
iii. Color proof
iv. Imposition proof
v. Sample/mock-up

All these proofs used to be analog on paper or carton, but in many cases they have been gradually replaced by online/virtual proofs. For the latter, a distinction should be made between data exchanges that take place via e-mail or through some special approval software. Approval via e-mail is much more cumbersome and error-prone than, for example, a method that is supported by portal software. It is crucial that the approval process is integrated into an MIS or the production controller or a workflow management system (see Section 2.4.2), because this is the only way to ensure that the right people receive the right proofs on time. Moreover, the software then can warn the PP if an approval request has been ignored or whether a proof has been approved or rejected. The system also tracks revisions. Thus, proofing should be integrated tightly into the workflow software. Let's also note in this context that approval delays are still common, which is why workflow automation in this area is a worthwhile undertaking.

In the following, let's discuss the proofs listed above in more detail.

(i) The visual inspection for technical properties of the PB's artwork is substituted by (or at least complimented with) automatic preflight programs. Nowadays, the print data is not only checked independently from the job order, but the job ticket is used to check compatibility with the artwork (colors, number of pages, etc.). The outcome of this preflight check might lead to an automatic correction, a notification of some operator in the print shop, or a message to the PB (e.g., an e-mail or an entry in a customer portal).

For the sake of completeness, I would like to mention quality inspection systems. They inspect the content of an artwork, for example, the expected legibility of 1D or 2D codes.

(ii) After preflighting, correcting, normalizing, color converting and trapping the PB's data, it is ready for imposition and creating the printing form. However, before the latter happens, sometimes an approval of the print-ready data is requested. In the simplest case, this can be done by sending a PDF file to the customer. This procedure has a crucial catch, however, because the default setting of the customer's PDF-reading software (usually Acrobat) can have a significant impact on the file's appearance. For example, the overprint preview can be switched on or off. Therefore, it is better to send rendered PDF data, i.e. image data, instead. However, the disadvantage of this method is that it involves a lot of data. One solution is to not send a file, but instead to store a preview image in screen resolution on the customer portal (or on some server to which the PB has access) into which the customer can zoom in as deeply as desired with dedicated software. Thus, only parts of the inspected image must be downloaded at higher resolution.

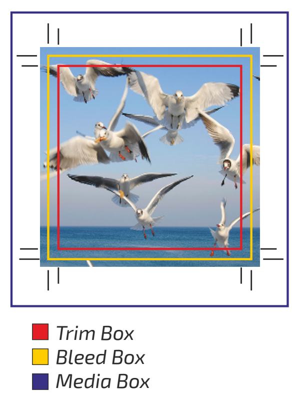

Figure 2.5: PDF Boxes

Of course, any checks that you can make in a PDF viewer must also be possible with this image-based method, such as the inspection of separations, display of total area ink coverage, measurement of color values, length measurements, showing trim, bleed, and media boxes, and so forth. Another feature of such page inspection software is the comparison of file versions and reshuffling of page positions in reading order.

It is fundamental that the PB can approve, reject and annotate a single page/artwork or an entire file. To accomplish this, the current approval status of each page/artwork must be indicated on the customer portal.

(iii) A *color proof* is a simulation of a print product in the color space of the planned press on the monitor. The goal is to anticipate the future print's color as closely as possible. Since this subject is very extensive, I will not attempt to go into the details of color proofs on inkjet printers or on monitors. That would be a separate topic involving special printing technologies, monitors, lights, calibrations, colorimetry, color profiles, color conversions, and the like, all of which would go far beyond the scope of this book on workflow automation.

Definition

An **imposition scheme** (often called **folding scheme**) indicates how pages are placed on a sheet. Only the principle placement of the pages in relation to each other is defined, not the exact position.

(iv) After the imposition process, the (PDF) forms used to be printed out on a plotter. The proof also showed the important dimensions of the trim box, the bleed box, and so on. A prepress operator then checked this *imposition proof* for mistakes in the position of the pages relative to the imposition template, missing bleeds, missing controls marks, etc. Sometimes, this proof was sent to the (internal or external) bookbinder to validate the imposition scheme. Today, only very critical impositions are inspected this way. Mostly, imposition proofs are omitted or reviewed on a monitor after having been rendered. This kind of proof is directed to print production experts only and not suitable for the PB. However, after de-imposing the imposed form – that is, splitting them into pages in the order according to the imposition scheme – the PB might browse through the pages in the reader's order. Sometimes, these are called *signature booklet*s. With these, a PB can check easily if the page layout on the sheet is correct.

(v) In the packaging industry, especially where cardboard/folding boxes are concerned, print buyers often request sample mock-ups. They can be of an analog or digital nature. Analog samples are made with cutting plotters. The digital version would be a 3D simulation of cardboard/folding boxes (see Figure 2.6). An animation allows visualizing and controlling the folding sequence.

For a *Request for Approval (RFQ)* and the actual approvals, the *PrintTalk* protocol can be deployed (see Section 4.5). The PP might send a *Request for Approval* business object, while the PB will send an *Approval Response* object. Both objects need to reference the job ID. To avoid misunderstandings, I would like to clarify (once again) that this of course does not mean that the PB and the PP write, send and read messages in PrintTalk, which is an XML instance. Instead, the software to which the PB has access (such as an e-commerce system or a customer portal) will send PrintTalk business objects when the PB clicks on appropriate icons in

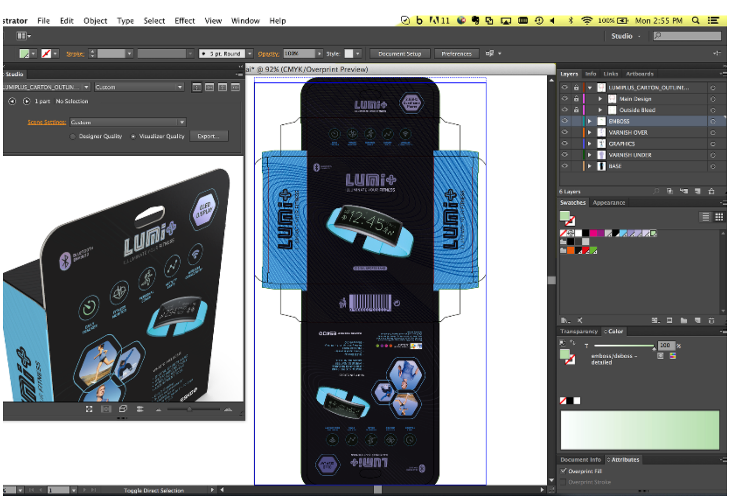

Figure 2.6: Virtual 3D mock-up of a folding box. Courtesy of Esko

the browser. Furthermore, the PP is not notified of the incoming message via SMS or some similar service, but the information is received by the MIS, for example.

During the production of a print job, there are other communications between PP and PB besides the approval procedure. The PP might notify the PB of a preflight report, the achievement of a milestone in the production cycle, or the tracking number of the shipment. Again, these notifications are either displayed on the customer portal or sent via e-mail. As an underlying protocol PrintTalk might be deployed using the business objects *Status Request* and *Order Status Response*.

During production, another message type might be sent from PB to PP, namely an order change. In theory, confirmed orders are immutable, but alas, it is rather common to forward job revisions.

2.3 Material Procurement

Automation in procurement is particularly worthwhile if many different kinds of material are required in small quantities. A web offset print shop that generally uses only a few different printing substrates and plates will have a framework agreement with the material supplier and thus will not need new quotes when material is ordered. The same can apply to a web-to-print shop. In the case of commercial offset printing with short print runs, on the other hand, it makes sense to consider automation in material procurement. In the following, I are assuming such a situation.

Materials management begins with determining whether there is enough of the required material in the print shop's stock. Only if this question is answered in the negative (by the materials management system) must negotiations between the print provider and the supplier of consumables be initiated.

The communication between many print providers and supplier of consumables is still very traditional regarding the purchase of printing substrates, plates, inks, or other materials. Often, some employee in the print shop's purchasing department calls the sales department at the consumables supplier, learns the current prices, and places the order over the phone or later by sending an e-mail/telefax. (Figure 2.7 - ❶) The consumables supplier will then feed the data into his MIS and forward the material to the PP, including the delivery slip. (Figure 2.7 - ❷) The invoice is sent by separate mail. (Figure 2.7 - ❶) The PP needs to manually type the data into the warehouse management system and/or into the MIS.

This procedure is particularly unsatisfactory for the calculation of a quote. Either the PP's employee has a price idea for the printing substrate, or he uses potentially outdated prices in his database (entered during the last order), or he must contact the paper supplier repeatedly. The latter is especially problematic when one considers that many quotations do not lead to a purchase order.

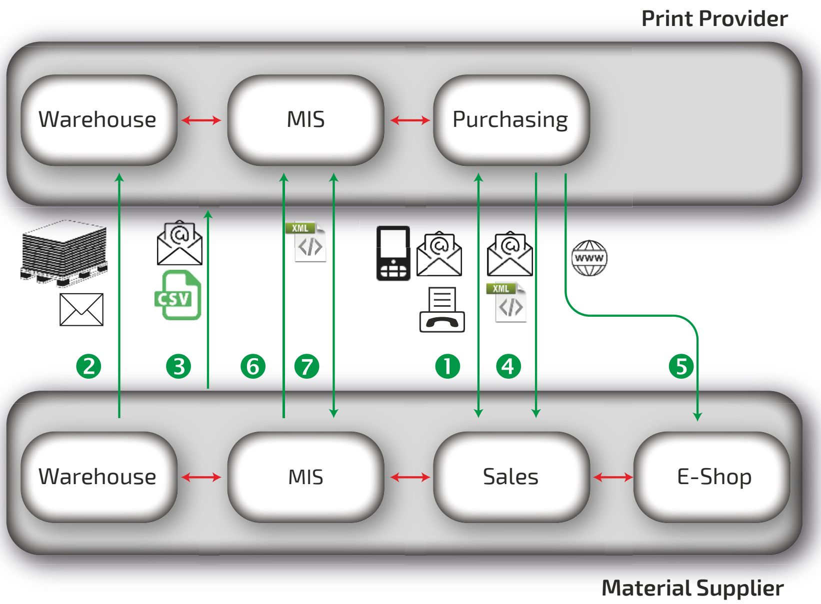

Figure 2.7: Communication between print provider and consumable supplier

There are some very special solutions for streamlining the workflow:

- The consumables supplier generates a price list from the MIS, for example as a CSV file on a regular basis, and sends it as an e-mail attachment to the PP. The PP imports the list into his MIS in order to make the currently valid prices available for the calculation of quotes. (Figure 2.7 - ❸). The PP can thus update the MIS database for a certain type of consumables (e.g., printing substrates) from this particular supplier.
- The PP sends orders to the supplier as e-mail attachments (in XML or some other text format). This might ease the import of the data into the supplier's MIS. (Figure 2.7 - ❹)
- The consumables supplier invests in an e-commerce system so that the PP can place orders online. This simplifies the processes for the consumables supplier, although the PP still has to enter the request manually. (Figure 2.7 - ❺)
- The PP scans delivery notes and invoices with OCR or automatically analyzes e-mails from the supplier in order to simplify the data input into his warehouse management system or his MIS.

All of these are very individual solutions for specific situations.

In the future, electronic invoicing might become more popular. The material provider's MIS will generate a PDF invoice, which embeds all relevant data (for example, as an XML record) in a machine-readable format (Figure 2.7 - ❻). The *ZUGFeRD* specification, for example, makes this possible (Forum elektronische Rechnung Deutschland, 2020).

All these solutions allow only some kind of semi-automatic workflows. The fully automatic solution, on the other hand, is that the PP's management information systems and the consumables supplier exchange XML data directly between their applications via a web service (Figure 2.7 - ❼). Printing substrates, for example, can thus be ordered fully automatically according to the scheduled printing date in line with the production schedule. Moreover, changes in dates, quantities and articles are also transmitted automatically. Right now, however, such exchanges are not yet standardized, and MIS systems of material vendors have different web service links (if they have one at all). Hence, the PP's MIS must support various formats.

Print providers normally need to communicate with many different consumables suppliers. With each of them, the method of communication may be different, which increases the overall complexity. Possible standardization could be achieved via platforms to which the PP's MIS connects (see Section 2.6).

> **Definition (by W3C)**
>
> A **web service** is a software system designed to support interoperable machine-to-machine interaction over a network.

2.4 Production

The production floor has traditionally been the pioneer in workflow automation. In the 1990s, first attempts were made to automate internal production workflows with the help of a standardized data format (see Section 4.2). At the turn of the millennium, the workflow automation effort was taken much further with the publication of the Job Definition Format (Section 4.3).

Figure 2.4 in Section 2.2.1 shows different paths connecting incoming orders to the production process. For example, there might be a specific method that the MIS uses to send job data to the production (often using the JDF or XJDF data format, see Sections 4.3 and 4.4). W2P orders, however, might be forwarded to the production floor via a different path. This could be caused by different production systems that are deployed for both areas.

There are efforts to unify these channels, for example with the help of an MIS or separate "middleware" software. This module would gather all the orders and print data, assign them automatically, and bundle the data for the following:

- Technical production processes

> **Note**
>
> **Middleware** is a type of computer software that mediates between applications. It can be described as "software glue".

- Orders to external suppliers
- Logistics companies
- Internal storage

This strategy reduces the number of interfaces and software modules within the production. It makes the production more streamlined and cost-effective. It reduces maintenance and increases overall production efficiency. For the latter, it is worth mentioning as an example that creating gang forms becomes more efficient the larger a job pool gets.

Of course, the single connection between MIS/middleware and production shown in Figure 2.8 is an idealized sketch because different printing technologies and print products require different processes and devices. They do not necessarily have to communicate with a universal production controller, which in turn exchanges data with the MIS, but may also be able to communicate directly via the MIS.

Many devices such as presses or postpress machines are individually connected with the workflow network. At times, a specialized controller governs these devices (see Figure 1.1 in Section 1). In prepress, however, applications often cover a few processes, normally under the direction of an operator (see next section).

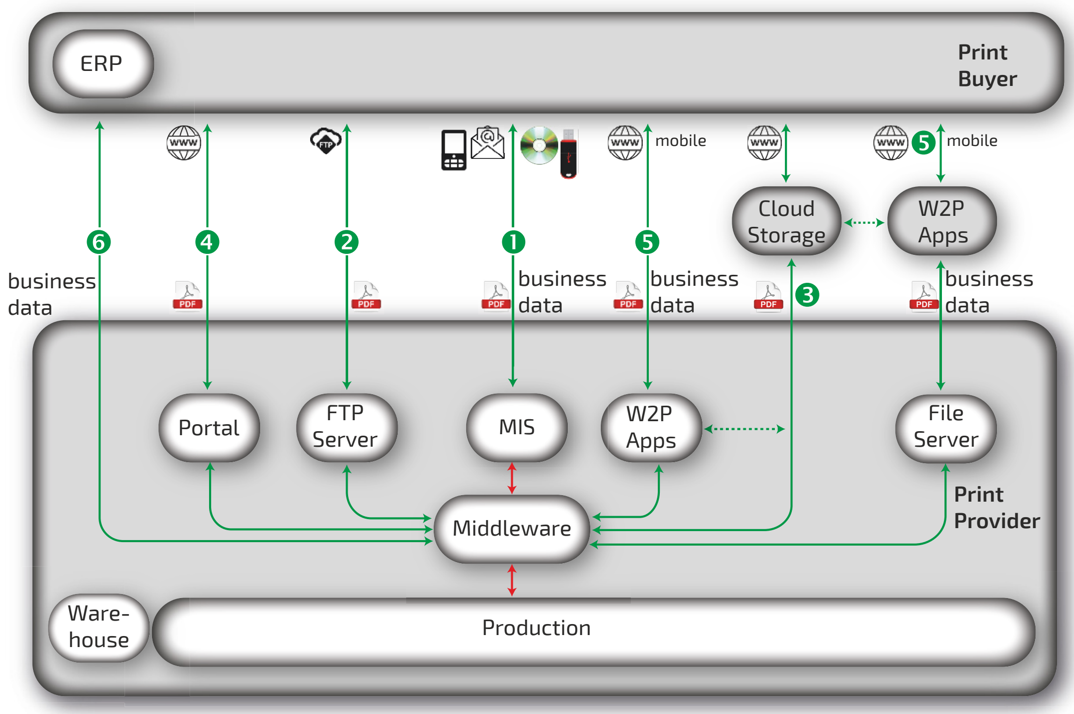

Figure 2.8: Communication between print provider and print buyer

A workflow management system (WMS), executes several different processes. Its software architecture is quite different compared to a stand-alone application. Let's discuss this in 2.4.2. The most networked module in the production environment is the electronic planning board (2.4.4). This software is, in a way, the champion of workflow automation.

2.4.1 Applications for Production

At first glance, it seems that every device and every software application is responsible for a single process. At second glance, however, numerous devices might come to mind that can execute more than one process. A striking example of this is a saddle stitcher, which collects and stitches. These are two different processes. Even a conventional sheet offset press can do more than print these days – it may also be able to varnishing, perforate, die-cut, emboss, and crease. However, it is still true that the main task of such a press is to print. The situation is different with a web offset press, where additional equipment such as a folder and a cutter is standard. Digital printing presses are also frequently supplied with inline finishing modules.

Nevertheless, it is still true for the press and postpress segments that, as a rule, a machine has only one main application, i.e. it executes a single process. In the prepress area, however, things are completely different. Here, too, there are still software programs that execute only a single process, such as preflight or color conversion. In recent years though, the functionality of individual applications has increased significantly. A typical preflight program nowadays can also fix errors of a PDF file, manually or automatically, and convert the file into some standard format (such as PDF/X or PDF/A). Imposition, trapping and color management can also be part of the PDF editor's scope of services.

About 20 years ago, raster image processors together with imposition software have developed into workflow solutions that can cover a wide range of tasks. Such software is often called a *workflow management system (WMS)* or just *workflow system*. Let's discuss them in the following section. The distinction between software application and WMS is somewhat blurred, however. The terms workflow management system or workflow system are now also used for workflow solutions in the press and postpress sector. At the same time, RIP technology has developed into Digital Front Ends (DFE) for digital presses.

2.4.2 Workflow Management System

Individual production processes are often handled not by isolated, stand-alone applications, but rather by a software system. Such a software system controls several modules/devices, and each of

them in turn can execute one or more processes. The entirety of all modules and the control software itself is called a *workflow management system* (WMS), the control software is called the *core*, and the individual modules/devices are called *workflow engines*.
A WMS is therefore not a monolithic application that fulfils many tasks, but rather a conglomerate of workflow engines that communicate with each other via interfaces and above all with the core of the WMS. The core initiates and controls the execution of the workflow engines (see Figure 2.9).

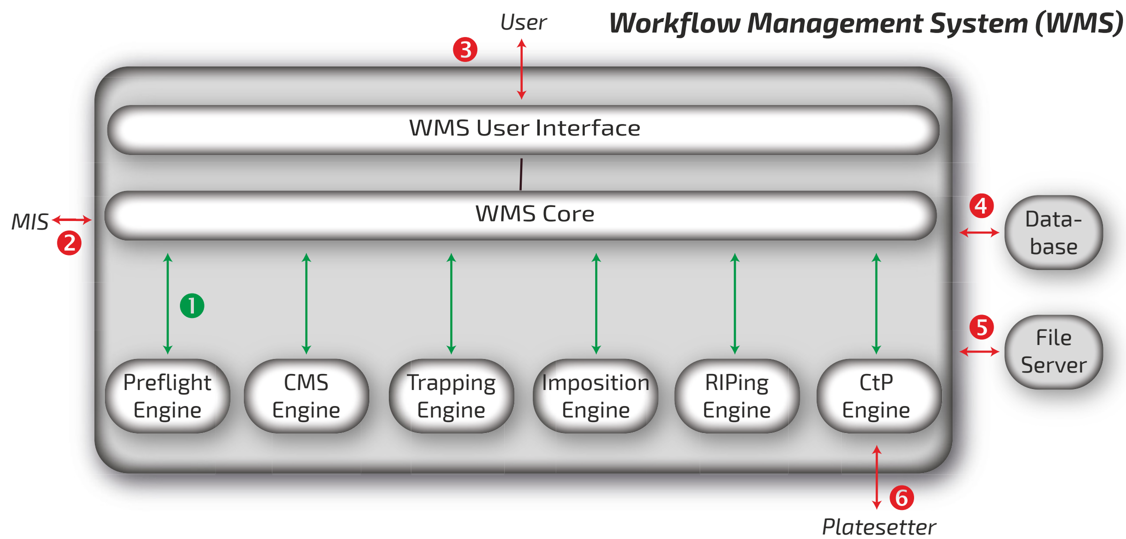

Figure 2.9: Interfaces of a workflow management system in prepress

> Note
>
> **CMS** stands for **color management system**. Color management is the controlled conversion of color between the color representations of various devices.

Three properties characterize a WMS:

- There is a common user interface for all workflow engines.
- A WMS administrator can model the process chains for various production processes within the framework of the WMS.
- A WMS administrator can define default values for the workflow engine parameters.

Generally, the workflow administrator uses a graphical user interface to compile visually different processes into a process chain. Arrows are used to connect the individual icons to specify the sequence and dependencies of the processes. Furthermore, process-specific property tables can provide the processes with instructions and default parameters. Figure 2.10 shows an example of a graphical representation of a workflow. This simple workflow example requires PDF and JDF files for each print job as input. It then sorts the PDFs according to the paper they will be printed on. This information is stored in the JDF file. The workflow management system in this example is *Switch* from *Enfocus*. The special feature of this WMS is that intermediate results between the indi-

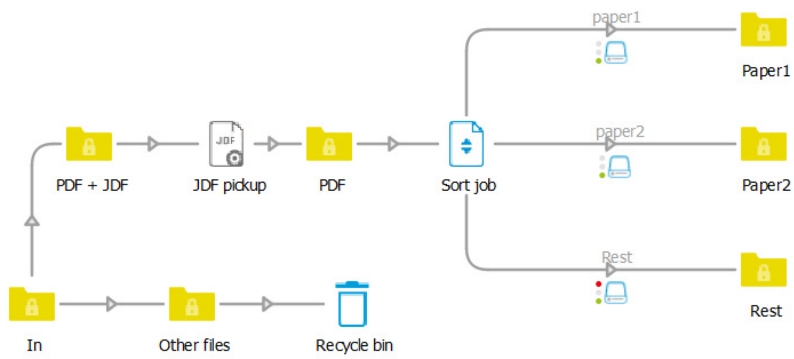

Figure 2.10: Designing a workflow with a workflow management system

vidual processes or phases can be accessed in graphical objects looking like folders, which makes it easy to check whether the processes are doing what they are supposed to.

The workflow core provides the workflow engines with necessary parameter data so that they can execute a process. The core in turn receives the data from the following

- External job tickets, for example issued by an MIS (Figure 2.9 - ❷)
- Result values from preceding workflow engines within the WMS (Figure 2.9 - ❶)
- Default values of the WMS set by a system administrator via the user interface (Figure 2.9 - ❸)
- Values that the WMS operator enters in dialog windows of the WMS (Figure 2.9 - ❸)
- Values extracted from the content data

When you use a WMS, there are at least two user roles: While the administrator sets standard parameters and defines the process chains for various production routes, a WMS operator will change predefined workflows and parameters in a few cases for an individual job only. Standard jobs can thus pass through a WMS completely automatically, at least in prepress. This is especially true for installations at W2P print shops.

Communication within the WMS (Figure 2.9 - ❶) can be handled by:

- Vendor-specific data formats (such as private XML)
- Standardized job tickets, such as JDF or XJDF
- A combination of the two

If standardized job ticket formats are used, a workflow engine is also called a *job ticket processor*. In theory, job ticket processors for a specific process are exchangeable if they are based on the same standard. Thus, during the installation there might be a choice of job ticket processors from several manufactures. This is not always true, however, and such a configuration must be analyzed and tested very thoroughly in advance. Unfortunately, a normal user cannot accomplish this – the manufacturer or some specialized integrator needs to be involved.

Concerning the IT infrastructure, the configurations of the WMS core and the workflow engines are mostly quite flexible. The core and the engines might be installed on one or on several different servers. This provides scalability. An engine can actually be installed outside of the PP's premises "in the cloud", with communication taking place via web services. Recently, many WMS vendors have begun offering their entire system as a cloud-based service (SAAS). There might also be several instances of an engine running simultaneously in order to boost performance.

The communication between the workflow core and the workflow engines is bidirectional, i.e., the engines send status information back to the core.

The data exchange between the WMS and the outside world is normally based on standard formats, such as JDF/XJDF in Figure 2.9 - ❷. This job ticket data would stem from an MIS or some workflow controller. It also might be connected to a portal system in which customers can manage their jobs, upload content data, check monitor proofs, specify correction requests, or give approvals. The information is forwarded to the WMS, and the content data is automatically routed to the print job. Conversely, a WMS can forward the production status to the portal to inform the customer about the production progress.

Figure 2.9 - ❻ represents the interface between the CtP engine of the WMS and the actual CtP device. The engine might forward TIFF-B images to the device, i.e., individual bitmaps for each color separation. Status and error messages might go the opposite way.

Note that a WMS often needs to access external data – either files from the file system (Figure 2.9 - ❹) or records from a database (Figure 2.9 - ❺). For example, the list of all current jobs in the system is usually stored in a database rather than retrieved from the JDF files, which would be possible but quite slow.

In press and postpress, a WMS might only support one type of process/device such as *printing* or *cutting*, respectively. Here the main task of such an WMS would be to translate the standard job ticket format into device specific instructions, balance the workload, and prepare the job offline. The latter allows job-specific settings such as cutting sequences to be defined outside of the actual device on a computer and thus not to block the much more expensive device during this time.

A WMS manages several workflows. From an abstract point of view, a workflow is managed internally by a system of rules, where a rule consists of events, flow controls, and actions.

Events are triggers like these:

- Job arrived in queue or hot folder or ftp-site or portal
- Approval has been granted
- Sheet is populated with content data
- Plates have been exposed

Conditional branches (if-then queries, loops, etc.) implement flow controls. Conditional branches can be refined by regular expressions.

Definition

A **regular expression** is a sequence of characters that defines a pattern.

Actions are activities, such as the execution of a process (by a workflow engine). However, an action could be also just a phase of a process or even just a basic task, such as:

- Extract information from data…
- Convert data into…
- Copy a file to…
- Send an e-mail to…
- Open a dialog box…

Each action receives a status after its completion (e.g., success, warning, error) that can be linked to other actions. That is, each action ends implicitly or explicitly in a flow control. In most cases, the WMS manufacturer predefines the actions, but more experienced users can also write their own in some environments (using programming languages such as *Visual Basics* or *JavaScript*).

Finally, the MIS, a production controller, or a WMS must be able to map print product types to different workflows. A business card, a folding carton box with a lid or a hardcover book will obviously require different workflows and different product parts. In fact, things are even more complicated: There are mandatory, optional and impermissible processing steps for each of these product types. Moreover, the production processes depend not only on the product type, but also on the printing technology. Producing a label on, for example, a flexographic press is different from producing a label on a digital press.

A WMS gathers operational data from its engines, edits and visualizes the data, and passes it to the MIS or to some other controller. The devices can compile different types of data:

- Messages – for example, the current state of an engine such as the current printing speed

- Reports/protocols – for example, a process summary per job after a process has been executed (how many good/bad sheets were produced in total, etc.)
- Machine control values – for example, the parameter settings of vacuum-powered pickup suckers for the feeder, and delivery or the actual ink zone settings

Messages can be used for progress bars or a status indicator on a dashboard. Protocols are the basis for controlling, reporting, and post calculation. Machine control values are useful for repeat jobs.

2.4.3 Operational Data

Note

Operational data is often also called **shop floor data**. Shop floor data collection is abbreviated with **SFDC**.

Operational data is sent from devices to a controller or from a controller to another controller, e.g., the MIS. There are four main reasons why operational data from production devices is important.

- Scheduling of jobs
- Status indication of production devices
- Job tracking
- Post-calculation and reporting

To schedule jobs, the planning board software needs to be up-to-date concerning the production status at all times (see 2.4.4). In order to be able to react instantly to any production disturbances, the status of each device should be continually known. In 2.2.3, we discussed that in certain configurations the print buyer might be able to check the status of his order, for example via a web portal. To ensure this, the devices must be able to inform the system which jobs they are currently processing and which they have just completed. Finally, to calculate the final cost of a job, data is needed from each device, such as consumed material and elapsed time. A management decision about the profitability of a product type or the efficiency of a device should be based on such data.

As with a WMS, there are two categories of operational data in the current context:

- Reports
- Messages

Reports are records of the result of a process or a process phase after it is executed. A report is static in the sense that it should not be changed later. it is a summary of the activities during the execution. Examples include total material consumption, time stamp at beginning and at the end, and any milestones that were reached. It should contain the final status after completion and can contain the name of the employee who was involved in the process.

The status should cover the device itself as well as the number of product components that were produced, separated into waste and good. It might also include special events during a process or a phase (such as a paper jam in the press) or extra costs that can be charged (for example, due to change of materials or requirements, correcting files, etc.).

In contrast to the reports, messages are issued dynamically. They provide information about the current status of a device, its material consumption, and the production progress. A message may signal the current processing speed or the current status of the process or process phase. In fact, messages can be issued without any process currently being executed on a device at all. They may just inform about the status of the machine (such as idle, breakdown, failure, busy, pause, or repair). The values in the messages can control progress bars and other graphical elements on a production dashboard. They are also used for updating a planning board.

Messages can be issued either spontaneously (e.g., if an error occurred) or at regular intervals (e.g., to provide information about the status of the device or its material consumption). The latter are usually issued without any hand-shaking protocol, sometimes also denoted as “fire and forget”.

A device can send messages either on his own or because it is told to do so by a controller. A controller can interactively instruct a device to send short messages on its own at regular intervals. The controller can specify an addressee in this request – either itself or another workflow component such as the planning board. However, it can also request information very specifically from a device. There can be several reasons to do this. For example, the controller might want to get information about the device’s capabilities. This can be very general or highly detailed. During a plug-and-play configuration of a workflow, a controller might just want to know which process a device can execute in general. However, the queries can also be much more in depth, such as querying a folder concerning the feasibility of folding paper of a certain size and quality according to a certain folding scheme. The implementation of device capacity protocols will certainly be one of the basic elements of autonomous print production in the context of Print 4.0.

Refining the Figure 2.1, Figure 2.11 is showing a more detailed data exchange between MIS and production systems. A similar figure could be drawn for the interface between a general workflow controller and a WMS.

A device might send the message to the kernel of a workflow management system, to some general workflow controller, or directly to an MIS. A WMS or a controller can forward this information, for example to the MIS or the planning board. It may summarize and/or shorten the content of the messages before doing so.

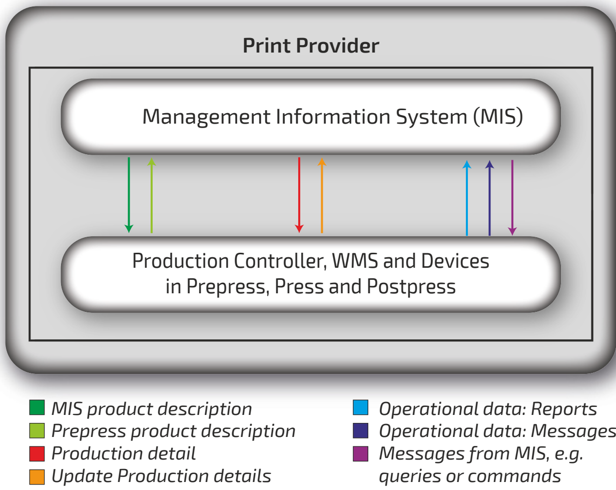

Figure 2.11: Interface between MIS and production systems

How is operational data collected? There are three options:

- Manually
- Semi-automatically
- Automatically

In modern print shops, it will be a mix of the last two. Manual gathering means in the worst case that each production operator will write on a printed form what he or she is doing. Next, some other person will collect the forms (for example, at the end of a shift) and enter the data into a spreadsheet, which is then used to analyze the production and to generate reports. An already significant improvement is when all operators enter the data into a dialog screen on a (standalone) computer next to the device, and the values are then automatically transmitted to an MIS, for example. Instead of entering data into a dialog screen, barcode scanners are also often used. This replaces the manual transfer of the data from a form to a computer program like Excel. Moreover, the data is gathered much faster, for example at the end of a print job. Finally, the data will be more accurate since it is entered immediately after having finished a process for a certain job and not in bulk at the end of the shift. I like to call this procedure “semi-automatic”. In either case, however, the focus is on reporting, messages are not really available.

Real-time messages are only feasible in machine-to-machine communication. The press can signal, for example, its current printing speed or the material that has been consumed so far. Another example of messages generated by a press concerns quality assurance during printing: A camera on the press captures images, and image processing software compares these with the original PDF artwork. Some types of messages, however, an operator may still enter manually, for example the reasons why he or she stopped a press.

Since not all machines can report operational data and not all processes run on a machine or some software application (examples include manual workstations in postpress), there is always the need to capture semi-automatic data within a fully automatic data collection environment. To calculate the actual costs of a product, it is essential to get the relevant data for all production processes.

In many cases, standard JDF and JMF protocols are used for data collection (see Section 4.3), but some manufacturers also use private protocols.

Finally, I would like to stress that not all the operational data of a device is sent back to the MIS. For example, a press might store certain data internally in order to support reprints. Inspection systems and machine sensors generate lots of data that will play an even more important role for the Print 4.0 concept. Analyzing such *big data* allows for the determination of error causes, predictive maintenance, quality enhancements, and generally better machine control. As an added benefit, the knowledge generated this way is not restricted to the direct machine environment. For example, a press could send data to prepress in order to place labels on a sheet in a more print-optimized way next time.

Printing presses may store their data on a central print server. The data exchange protocols used for this differ from the ones we are covering in Chapter 4. The machine-to-machine protocol OPC-UA (OPC Foundation) is used in these industrial applications.

2.4.4 Planning Boards

Often, an MIS also schedules the chronological order of jobs, i.e., the job queue for each device. The overall objective is to synchronize and determine the production sequence of orders depending on their characteristics and production-related restrictions. A possible variant of this is that the MIS transmits all job data to a separate planning tool (*scheduler*) or to a workflow/production

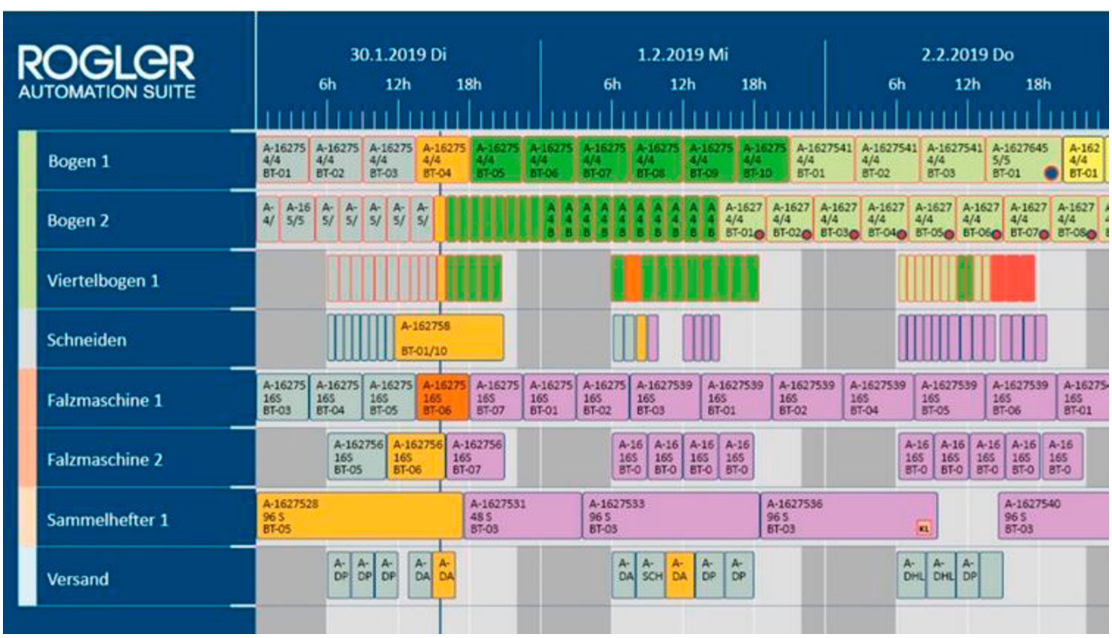

Figure 2.12: A planning board shows the production steps and the status of all current print orders. Courtesy of Rogler AG

Note

Gantt diagrams (charts) show the dependency relationships between activities and the current schedule status. They are often used in project management.

controller. The graphical interface of the software displays a planning board that gives the user in the print shop an overall view of the production and planning status in real time. Figure 2.12 shows recent, current and upcoming production steps for orders in a Gantt diagram.

An improved scheduling system can increase the overall productivity of a print shop tremendously. In connection with an offset press, for example, changeover times between jobs (de-inking, blanket wash sequences, changing paper size at the feeder and delivery units, etc.) and the waste of printing substrates can be reduced. Another reason for the importance of scheduling is that the prerequisites for the execution of a production process are triggered in time. For example, it must be ensured that a platesetter images all plates for the printing process in advance. To do this, other processes must have already been executed (preflight, color management, trapping, imposition, and RIPing). Another advantage of a well-organized production sequence is the avoidance of unnecessary breaks between processes. However, required drying times must be taken into account in some cases. Overall, digital planning software delivers a faster production throughput and cost savings, especially in the postpress area, as waiting times and unnecessary material movements to and from the intermediate storage facility can be avoided. Above all, a balanced and smart allocation of orders to the production machines will lead to higher machine utilization.

Ideally, the planning software sends to each device on the shop floor an individual order list that displays the execution sequence. The order list must be synchronized across all devices, of course, to make sure that the outcome of each process is available for the next process according to the schedule.

Another highly appreciated feature of an electronic scheduler is flexibility. Last-minute production changes due to a breakdown of a machine or some reprioritization of a job, for example, no longer have to be initiated manually. The scheduling software will dynamically adjust to the new situation and change the schedule on its own.

An electronic scheduler can be viewed anywhere in the print shop, giving everyone an overview of all planned and executed jobs in real-time. In particular, missed deadlines will be transparent.

Most schedulers have different modes: automatic, semi-automatic, and manual planning. You can manually overwrite planning data that had been generated automatically. Characteristics of production planning can be adjusted, for example by assigning the highest priorities based on deadlines, set-up time optimization, or machine utilization (these constraints might conflict). AI algorithms are also applied to determine the best production sequence. Moreover, you can define shifts and block time intervals for recurring jobs.

A scheduler must be connected to the order administration software (for example, to the MIS/ERP) as well as to the devices in the print shop and the *warehouse management software*. The MIS provides product and production descriptions (JDF or internal data formats) as well as deadlines to the planning software. The latter forwards (JDF) jobs to the devices in production schedule order. The orders are placed in sequence depending on the equipment and the required consumables. Vice versa, the production devices send status and material consumption information to the scheduler. The schedular must evaluate operational data (both reports and messages – see Section 2.4.3) in order to display production steps/milestones on a dashboard and the current job statuses on a timeline. This makes a scheduler the most prominent hub of workflow integration. In addition, it must of course have precise knowledge of the individual devices and the interdependencies between processes in order to be able to optimize the job sequence and thus achieve a reduction in the transition time between jobs. Obviously, schedulers are also used for partial areas of production, such as for the printing process only.

2.4.5 Warehouse Management

Although warehouse management and fulfillment are often thought of as a single issue, I would like to separate the two and understand fulfillment as the dispatching of goods to the print buyer. We will deal with that in the next section.

A print shop warehouse usually stores more than just finished products, namely printing substrates, consumables such as ink or plates, spare parts, semi-finishes products, end-products, merchandise, and packaging materials.

Probably the most important feature of a warehouse management system involves recording goods in and goods out, for example by using barcode scanners or simple input dialogs on mobile devices. Every material movement should be entered into the system. A proper warehouse management system must also know the storage capacities and be able to autonomously find a suitable place for an incoming item. The storage location of an item is recorded in a database from which it can be retrieved at any time. When items are moved from one place to another within the warehouse, the warehouse software records the movement.

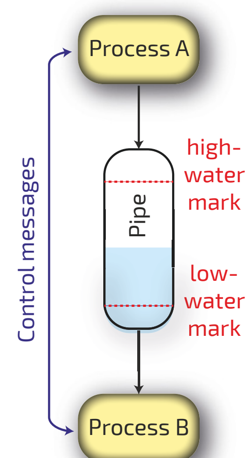

Figure 2.13: A pipe (buffer) with low-water and high-water marks

A warehouse management system can also warn of overstocking and understocking. An analogy of a *pipe* illustrates this process (see Figure 2.13.) The entry of an item into the warehouse means that it is filled into a pipe. When the item is taken out of the warehouse, it is taken out of the pipe again. This can happen in parallel, overlapping, and successively. As a rule, it is important that the inventory level in the pipe does not fall below a specific low-water mark, as this would prevent the continuous removal of items on

Note

A model of a **pipe** is uni-directional and handles data in a first-in, first-out fashion (FIFO).

the other end. If the article is a semi-finished product or the end product, falling below the low-water mark might indicate that the requested number of units is not available. Of course, the level of the pipe should also not exceed a certain high-water mark, because then the storage capacity may no longer suffice and the warehouse may be filled with excess items, which ties up capital, signals overproduction, and generally increases storage costs. Each article has its own pipe and own high-water and low-water marks that must be set individually.

The water marks should be modified dynamically, for example in accordance with the production forecast if the article is some raw material. If it is a semi-finished product or the end-product, the requested number of copies of the print order(s) or some forecast relating to this should control these marks. In both cases, the MIS needs to feed the data to the warehouse management software.

If the level of an item falls below the low-water mark or exceeds the high-water mark, appropriate departments must be notified. For example, if the level of a print substrate falls below the low-water mark (the minimum stock level), the material procurement department should be notified. It is conceivable, of course, that this will also trigger and automatic order to material supplier.

The MIS should book raw material as early as possible, for example when a quotation becomes a print order. During production, the MIS should also ensure that the material is retrieved from the warehouse in a timely manner. Finally, the MIS should trigger the transfer of the final products to the fulfillment department. This is particularly true if a consignment store is run.

The CIP4 organization has defined the mechanism of dynamic pipe control in its specifications irrespective of any warehouse storage. If an output resource of process A is the input resource of process B (see Figure 2.13), the JMF/XJDF pipe control mechanism might be deployed. However, this does not necessarily mean that the resource is brought into the warehouse and then retrieved again. It could be buffered on a conveyor belt or simply somewhere on the shop floor.

A warehouse management system can be very powerful in connection with an automatic transport system and automatic paper feeding of a press. But even without it, the internal transportation of material on shop floor can be simplified significantly, for example by installing a barcode system for pallets. The software creates a pallet label for each pallet with a unique barcode that is scanned at each workstation. In this way, the system knows the locations of all pallets and can create transport orders independently. To create such orders, it must be able to communicate with the planning board (MIS, production controller) since this the only entity that knows the order of the production steps.

Similar to the planning board, the warehouse management software can be an optional module of an MIS or stand-alone software. I would like to mention in this context that warehouse management software is often browser-based or app-based nowadays, supports mobile devices, and can run on-premises or as a cloud-based solution. For the latter, the communication between client and server should always be encrypted.

2.5 Fulfillment

At first glance, fulfillment may seem simple, but alas, it is not. Let's look at some of the complexities involved in executing a single job:

- Multiple products, shipped on palette(s) or in parcels
- Multiple delivery addresses
- More than one shipment
- Different product quantities to various addresses
- Multiple departure times for the shipments
- Different packaging requirements for the shipments (local, international)
- Various procedures for the shipments, such as custom declarations for international shipments
- Different carriers for the shipments
- Addition of pre-produced or externally produced products from the warehouse

The range of automation in delivering and transport is huge. For example, a prepress operator could simply ask a local bike courier

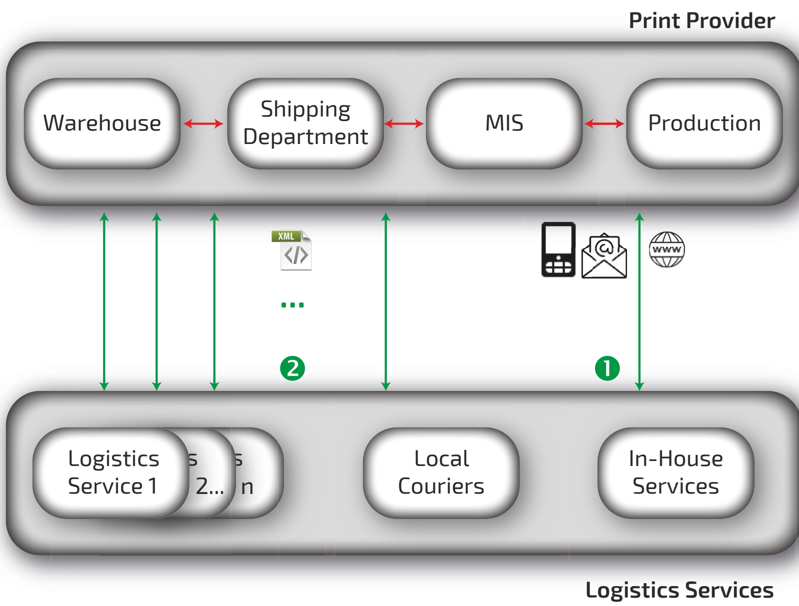

Figure 2.14: Communication between print provider and logistics services

service to deliver printed proofs to the customer (Figure 2.14-❶). On the other hand, there are highly automated fulfillment solutions.

A very traditional way of sending printed products is that an MIS – or even just some text processing software – prints a mailing list, address labels and palette labels on a (label) printer. Especially if a single printed matter is shipped to numerous customers, it is quite convenient to use a list of addressees that the PB has provided in a table. The employees in the shipping department pick the individual items, pack them, label them, and tick them off on the list.

Shipping software, which is often a module of the MIS, can operate more efficiently. For example, the mailing list is stored digitally, and filling the list is done with the help of a barcode scanner. As soon as all required items are registered, the delivery bill is automatically created, printed, and forwarded to the billing address. As soon as the loading of a truck is completed, the software prints the loading list automatically.

Packing and palletizing may be done either manually by an operator, semi-automatically, or fully automatically by robots. Robots are mainly deployed if the print shops processes jobs with high print runs. Of course, the packing of personalized bulk deliveries is completely different and consists essentially of enveloping and addressing.

Special attention should be paid to the interface between the PP's shipping software and the logistics provider that transports the goods.

The aim is to send the shipping data to the logistics company electronically and to have the price and the tracking numbers returned in the opposite direction. The latter can then be used to create an automatic delivery notification including the tracking number for the print buyer. Moreover, accurate package labels with all necessary information for the carrier need to be printed in the right format. Again, web services can be deployed to establish a direct communication link between the shipping software and the logistics carrier's system (Figure 2.14 - ❷).

As already mentioned in the Materials Procurement section 2.3, each company usually has its own API regarding the use of web services. Since there are hundreds of different companies working in the logistics sector, it is difficult for a fulfillment software manufacturer or an MIS provider to support all of them. While everyone knows USPS, DHL, UPS and FedEx, there are many others (consigner, 2021). As a result, they will only be able to manage a few of them, which is why there are companies that specialize in reconciling the different API interfaces of the logistics carriers with a universal interface for the shipping software/MIS (Figure 2.15 - ❶). A distribution software/MIS producer can then license the logistics

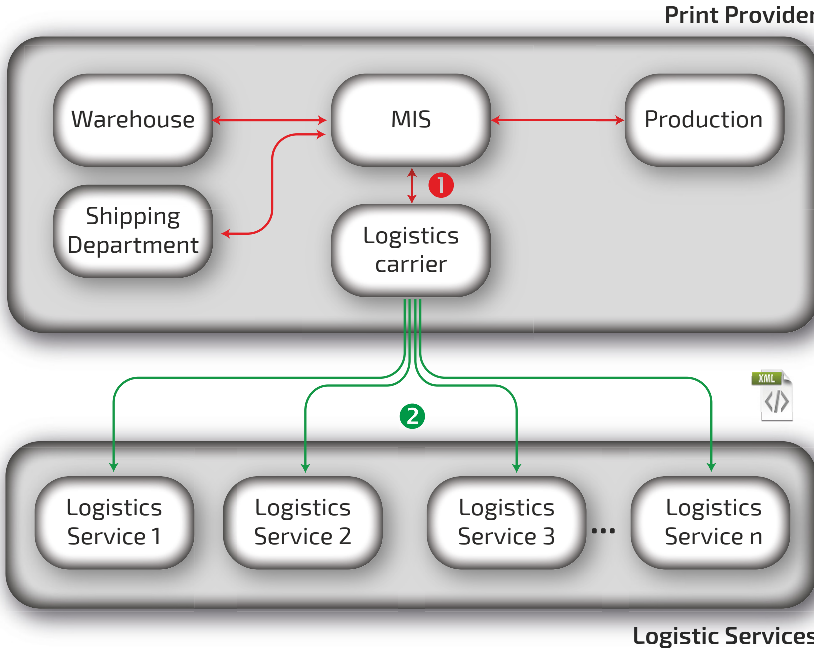

Figure 2.15: MIS/Shipping software with 3rd party module for data exchange using Web services

carrier software as some kind of middleware between it and the actual logistics carriers.

The interface between the shipping software/MIS and a specific logistics carrier module can be very simple. For example, the communication channel might consist of just two hot folders (one for each direction) in which text files are transferred. The text will be structured, of course. For example, CSV, XML or JSON files are used. If the MIS does not incorporate the shipping software and an external software is deployed instead, a similar interface is needed between the two. The data exchange ❷ in Figure 2.15 is often based on web services.

> **Definition**
>
> The text-based, structured data format **JavaScript Object Notation (JSON)** is similar to XML.

The logistics carrier module may or may not provide an automatic calculation to determine which carrier is the most reasonable company for sending a particular shipment. However, individual rules can be set up for choosing the carrier. Particularly if the shipments involves multiple countries, it becomes time-consuming and error-prone to work out the optimal delivery method for each shipment yourself.

2.6 Subsidiaries, Print Brokers, Subcontractors

Managing orders from subsidiaries of a print shop, forwarding orders from print shop to print shop, or communicating with subcontractors and print brokers have one thing in common: There is a need for forwarding job tickets or parts of them from one stake-

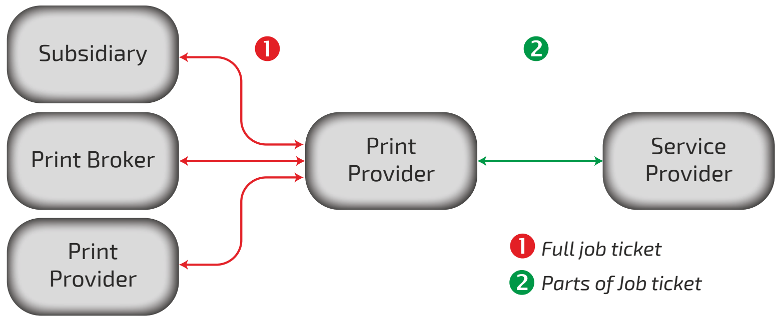

Figure 2.16
Job ticket exchange between print provider and partners

holder to another. Often, this also includes content data. Technical descriptions of the services and certain production data must be communicated, for example, if a print shop orders from a subcontractor embossing dies and die cutting tools or even complete production steps such as exposing plates, laminating, or bookbinding.

Theoretically, the concept is simple. A subsidiary, print broker or some other print shop sends a full job ticket with the description of the product details (e.g., PrintTalk with XJDF or simply CSV) to the print provider (Figure 2.16 - ❶). In a different scenario, a print provider sends parts of a job ticket containing product information as well as necessary production details to a service provider (Figure 2.16 - ❷). The PP in the first case respectively the service provider in the second case will ideally import the data automatically into his own system, produce the product or some component of the product or a production tool, and then might send (parts of) the updated data back.

Please note that there might be two different independent software systems involved, namely the MIS of the sender and the MIS of the receiver. Even if they are using the same software, it might not be trivial to connect the two. Only if a subsidiary and the head office both make use of a single instance of an MIS in the cloud can they share data easily by using a common database as long as MIS has been programmed to support multiple print shops.

In most cases, it not too hard to implement sending/receiving entire job tickets between A and B automatically in a secure way. This is especially true if the complete job ticket data exchange between branch offices or between a print broker and a print provider involves large companies, which can set up these connections. We already mentioned in 2.1.3 that these big (online) print shops develop their own MIS in some circumstances using the functionality of some commercially available MIS as a base (via API).

Things become much more difficult if a PP must extract some parts of a job ticket for a service provider and forward the data to him automatically. It is even more complex for the service provider to automatically import those job ticket parts from different print providers. Therefore, such data exchanges are still mostly done via e-mail, phone, ftp and the like in the graphic arts industry. In some cases, there might be customized solutions between two partners, for example via self-written scripts.

Since the job tickets are forwarded electronically, it is important to encrypt the transmission, whereas the data is often physically stored in a virtual private cloud (VPC).

Thus, the industry uses human-readable data (e-mails, etc.) and machine-readable data (CSV, XML, JSON, JDF, XJDF, PrintTalk) for the job ticket transmissions between stakeholders. However, even machine-readable data can have different levels of automation. For example, a CSV file could be assembled manually. Even if the exchange file is created automatically, for example by an MIS, the question remains whether

a) the data is automatically forwarded to the recipient (and not through an e-mail attachment) and
b) whether the recipient can automatically import the data back into his system.

Lately, online platforms have emerged which connect participants such as print buyers, print providers, and partners, e.g., suppliers of consumables, machine manufacturers, and service providers. To achieve this, any software connected to the platform can store product and production data on it. The platform will pass this data on to suitable partners, who will then receive a notification of it.

Definition (W3C)

A **virtual private cloud** (**VPC**) is a private cloud within a public cloud environment.

References

Forum elektronische Rechnung Deutschland (FeRD) (2020), ZUG-FeRD 2.1 1 Specification. Available at: https://www.ferd-net.de/standards/zugferd-2.1.1/index.html (Accessed: 15 June 2021).

OPC Foundation, https://opcfoundation.org/about/what-is-opc/ (Accessed: 15 June 2021).