



FLUJO DE TRABAJO JDF

GUÍA DE AUTOMATIZACIÓN PARA INDUSTRIAS GRÁFICAS

THOMAS HOFFMANN-WALBECK Y SEBASTIAN RIEGEL
TRADUCCIÓN MIGUEL EDUARDO MENÉNDEZ ROMERO

Autores Thomas Hoffmann-Walbeck y Sebastian Riegel

Flujo de trabajo JDF

Guía de Automatización para Industrias Gráficas

Traducción Miguel Eduardo Menéndez Romero

Copyright 2009

Thomas Hoffmann-Walbeck y Sebastian Riegel, jdf@hdm-stuttgart.de.

Versión publicada previo acuerdo con Hoffmann-Walbeck y Riegel

Copyright 2013

Traducción y edición castellano/español miguel menéndez asesor gráfico

Impreso en 2013 por PIA y miguel menéndez asesor gráfico

en colaboración con CIP4. Todos los derechos reservados

Depósito legal: A 180-2013

ISBN: 978-84-616-3926-7

Está prohibida la reproducción por cualquier medio sin permiso expreso.

Las marcas registradas por sus propietarios se mencionan en este libro a nivel informativo sin reconocimiento alguno por parte de PIA o miguel menéndez asesor gráfico

Los libros publicados por PIA son utilizados por particulares, empresas, asociaciones y escuelas de negocio de todo el mundo. Puede obtener descuentos por volumen contactando con:

Printing Industries Press

Printing Industries of America

200 Deer Run Road

Sewickley, PA 15143

Phone: 412-741-6860

Toll-Free: 800-910-4283 x770

Fax: 412-741-2311



Email: membercentral@printing.org

Online: www.printing.org

miguel menéndez asesor gráfico

email: info@miguelsenendez.com

Online: www.miguelsenendez.com

Prólogo

¿Por qué JDF?. O mejor dicho ¿por qué es necesario escribir un libro sobre el flujo de trabajo JDF? ¿No es simplemente un formato que transmite los datos correctamente a la máquina de imprimir?. Cuando usamos un teléfono ¿a quien le importan los protocolos que utiliza en segundo plano? o ¿acaso no podemos conducir un coche sin conocer los detalles técnicos de sus pistones?

El uso del Formato de Definición de Trabajo JDF es muy interesante para los especialistas porque describe la mayor parte de procesos de la industria de la impresión y representa un modelo importante para la industria de la comunicación gráfica.

La tecnología todavía es bastante nueva y aún no funciona de un modo explícito, como un coche o un teléfono. Instalar, configurar, probar y depurar un flujo de trabajo JDF requiere de ciertos conocimientos básicos sobre el formato.

Pero nuestra intención no es profundizar en el formato de datos. Consideramos más importante la construcción del flujo de trabajo con JDF. Por lo tanto, en los primeros capítulos describimos el proceso en sí mismo, con el fin de explicar su configuración con JDF posteriormente. Hemos tratado de separar las dos partes por si desea ver el proceso y la configuración independientes el uno del otro.

Este libro está dirigido a aprendices de la industria de la comunicación gráfica, estudiantes de impresión y tecnologías de la comunicación, impresores, fabricantes e informáticos.

Presuponemos que no tienen conocimientos de JDF, pero sí cierta experiencia en procesos de producción para manufacturar productos impresos. Por si fuera necesario, hemos definido algunos de estos procesos en el glosario para facilitar la comprensión.

Naturalmente, en un libro como este no podemos describir al detalle todos los pasos productivos para realizar un impreso. Existen libros de texto especializados en el procesado de imágenes, la creación de planchas, impresión offset, etc. Solo entraremos parcialmente en los detalles técnicos y lo haremos mediante ejemplos. Sin embargo, no queremos filosofar sobre un uso general, nuestro deseo es exponer claramente el proceso.

La intención es ilustrar los principios fundamentales del flujo de trabajo JDF. La divulgación de consejos y trucos es responsabilidad de las publicaciones comerciales especializadas en soluciones patentadas por los distribuidores.

Durante muchos años, hemos enseñado teoría y práctica en la Universidad de Stuttgart y podemos concluir que la literatura especializada en flujos de trabajo es exigua. La única documentación conocida son las especificaciones JDF [1,3] y [1,4] destinada principalmente a informáticos y desalentadora para el usuario no experto. Por otro lado, podemos consultar en los ejemplos [32] o [33] las publicaciones divulgativas sobre beneficios económicos de la automatización JDF.

Los fabricantes de soluciones disponen de buenos manuales sobre el uso de sistemas JDF, pero los libros sobre el tema son bastante escasos (por ejemplo [16]) y la mayoría solo describen segmentos de la producción [10] y [22].

Creemos firmemente que ha nacido una forma de producción para la industria gráfica. También sabemos que todavía está en desarrollo y que es aterradora para unos y fascinante para otros. Pero aprovecharemos esta edición para examinarla profundamente. Siempre podemos citar las palabras de Klimsch en su anuario de 1924/25 (página 109).

El desarrollo económico, despiadado e incontenible va por su camino.
Lo antiguo se desmorona y surge nueva vida de las ruinas. Una mirada atrás muestra la impetuosa evolución del arte gráfico en la última década.

En la introducción del capítulo 1, exponemos características y expectativas generales de un flujo de trabajo JDF. También comentamos brevemente el desarrollo que nos ha conducido a este formato. Quien haya estado al tanto últimamente en materia de JDF puede saltarse este capítulo.

En el segundo capítulo describimos los escenarios de tres imprentas reales. Son tecnológicamente muy avanzadas, pero solo dos emplean JDF. Los lectores necesitan conocer la diferencia que implica la inclusión de JDF. En la sección 4 aclaramos los conceptos específicos como el flujo de trabajo o los partes de trabajo y en la sección 5 describimos las características generales.

El modelo proceso/recurso o producto/consumidor es el tema del capítulo 3, donde abordamos las características básicas con ejemplos sobre las áreas de gestión de pedidos, preimpresión, impresión y acabado.

En el capítulo 5 exponemos XML brevemente, el Lenguaje de Mercado Extensible necesario para comprender el código JDF. En el capítulo 6, se presentan las estructuras JDF más importantes.

El capítulo 7 trata sobre el "Formato de Mensaje de Trabajo" (JMF), el "SMS de la Industria Gráfica". Se trata de un formato de datos y protocolo de comunicación en entornos JDF. Ambos cubren los detalles del flujo de trabajo y sus equivalentes JDF en las áreas de gestión de pedidos, preimpresión, impresión, acabado e impresión de embalajes y son la base de los capítulos 8 a 12.

En el capítulo 13 se analizan dos posibles proyectos JDF que pueden ser interesantes para el lector. En la primera parte se expone la implementación de un flujo de trabajo JDF en una imprenta. En la segunda sección ofrecemos una breve introducción a la programación de aplicaciones JDF con JAVA, con la única finalidad de proporcionar una orientación inicial sobre como integrar bibliotecas.

Al final de algunos capítulos hemos añadido ejercicios que ayudan a comprender mejor las materias.

Los autores desean agradecer sinceramente a las siguientes personas su valioso apoyo en la redacción de este libro:

- Mr. Dieter Adam (MB Bäuerle)
- Mr. Jan Breithold (HELL Gravure Systems)
- Mr. Ruben Cagnie (EskoArtwork)
- Ms. Anja Dannhorn (Fujifilm)
- Mr. Gottfried Grasl (Heidelberger Druckmaschinen)
- Mr. Stefan Kopec (Druckerei Mack)
- Ms. Ulrike Kurz (MBO)
- Mr. Bernd Laubengaier (Druckerei Laubengaier)
- Ms. Prof. Dr. Christa Neß (Hochschule der Medien)
- Mr. Lieven Plettnick (EskoArtwork)
- Ms. Ulrike Seethaler (Heidelberger Druckmaschinen)
- Mr. Matthias Siegel (MB Bäuerle)
- Mr. Klaus Stocklossa (MBO)

Autores del texto:

Thomas Hoffmann-Walbeck y Sebastian Riegel
c/o Hochschule der Medien
Nobelstraße 10
D-70569 Stuttgart
0711-89232128 or 0711-89232115
jdf@hdm-stuttgart.de

Autor de la traducción

Miguel Eduardo Menéndez Romero
info@miguelsenendez.com
<http://www.miguelsenendez.com>

Nota del Editor: Los números mostrados en el texto entre paréntesis son referencias a la Bibliografía de las páginas 205–206.

Contenido

Prólogo

1 Introducción	1
1.1 El Desarrollo de JDF	1
1.2 Características del Formato de Definición de Trabajo	3
1.3 Implementación de un Flujo de Trabajo JDF	6
2 Conceptos Básicos del Flujo de Trabajo	9
2.1 Empresa sin Flujo de Trabajo JDF	9
2.2 Empresa con Integración Parcial de JDF	11
2.3 Empresa conectada en red mediante JDF/JMF	12
2.4 Definiciones	13
2.5 Clasificación de los Flujos de Trabajo	16
2.6 El Formato JobTicket y las Propiedades del WMS	20
3 Modelos de Flujo de Trabajo de Impresión	25
3.1 Gestión de Pedidos	27
3.2 Flujo de Trabajo de Salida en Preimpresión	28
3.3 El Proceso de Impresión Offset de Hoja	34
3.4 Modelo de Ejemplo para Postimpresión	37
4 Historia y Aplicación de los Metadatos	39
4.1 Metadatos para Fotos y Documentos	40
4.2 Formato de Impresión de Producción (PPF)	45
4.3 Formato de Orden de Trabajo Portátil (PJTF)	50
5 Breve introducción a XML	55
5.1 Construcción de documento XML	55
5.2 Espacio de Nombre XML	57
5.3 Marco de Descripción de Recursos (RDF) y XMP	59
5.4 Lenguaje de Marcado Extensible para Comercio (cXML)	61
6 Introducción a JDF	63
6.1 Construcción de un documento JDF	63
6.2 Ejemplos de Nodos JDF	67
6.3 División de Recursos	72
6.4 Cajas Grises y Procesos Combinados	74
6.5 Arquitectura del Flujo de Trabajo JDF	77
6.6 Separar y Combinar	81
6.7 Especificación de Conformidad e Interoperabilidad (ICS)	83

7 Formato de Mensajería de Trabajo JMF	89
7.1 Modelos de Comunicación	89
7.2 Familias JMF	91
7.3 JMF ICS	98
8 Sistemas de Gestión de Órdenes	101
8.1 Funciones básicas de un Sistema de Gestión de Pedidos	102
8.2 Entornos de Producción	104
8.3 Documentos ICS MIS	116
8.4 PrintTalk/JDF Interfaz con el Cliente	118
9 Preimpresión	123
9.1 Interfaz entre el MIS y Preimpresión	124
9.2 Montaje	130
9.3 Sobreimpresión	135
9.4 Rasterizado y Producción de Planchas	137
9.5 Pruebas y Aprobaciones de Prensa	141
10 Impresión	145
10.1 Impresión Convencional	146
10.2 Impresión Digital	160
11 Postimpresión	167
11.1 Corte por Guillotina	169
11.2 Plegado	171
11.3 Alzado y Grapado	174
12 Impresión de Embalajes	179
12.1 Diseño y fabricación del Troquel y Optimización del Pliego	182
12.2 Perforado y Encolado de Cajas de Cartón	185
13 Proyectos JDF/JMF	193
13.1 Implementación de Flujos de Trabajo por Módulos	195
13.2 Flujo JDF con Módulos de Múltiples Proveedores	197
13.3 Programación JDF/JMF	200
Bibliografía	205
Glosario	207
Abreviaturas	210
Índice	214

1 Introducción

El “Flujo de Trabajo JDF” en la industria gráfica se entiende generalmente como el proceso integrado por los dos formatos normalizados, el **Formato de Definición de Trabajo** (JDF) y el **Formato de Mensaje de Trabajo** (JMF). Su finalidad es la automatización del proceso mediante la integración de los diferentes sistemas y aplicaciones. Por lo tanto, la idea básica subyacente es bastante simple: resumir la información de un trabajo de impresión y transmitirla a las partes interesadas que lo requieran. Si todos los datos necesarios se insertan en un solo archivo, es decir el tamaño de hoja, presupuesto, preimpresión digital, controlador de prensa, guillotina, etc y se transmiten a través de la planta, obviamente, reduciremos los costes.

Sin embargo, el flujo de trabajo JDF tiene otros objetivos que veremos paulatinamente, pero inicialmente, pueden servir como ejemplo la transparencia en los costes, la reducción de errores y el ahorro de tiempo.

1.1 El Desarrollo de JDF

Este formato fue iniciado por el fabricante de prensas Heidelberg, con la colaboración de ManRoland, Agfa y Adobe. En septiembre del año 2000 encargaron su desarrollo a la **Asociación de Cooperación Internacional para la Integración de Procesos en Preimpresión, Impresión y Postimpresión** (CIP4) creada a tal efecto. Se anunció y presentó en la Feria Drupa del mismo año. Actualmente la asociación CIP4 tiene más de 300 miembros, entre los cuales encontramos proveedores, usuarios de software, consultores e instituciones de industrias y artes gráficas.

La Organización CIP4 surge del Consorcio CIP3 fundado en 1995. CIP3 es la abreviación de **Cooperación Internacional para la Integración de Procesos en Preimpresión, Impresión y Postimpresión**. El **Formato de Producción de Impresión** (PPF), publicado por CIP3 también fue adoptado por CIP4. La sede de CIP4 está en Zurich, Suiza.

PPF y JDF/JMF son formatos de interfaz para soluciones de redes. Ambos se utilizan al inicio de la producción industrial de productos de impresión, a diferencia de la producción manual utilizada en el pasado. La terminología que mejor lo define es **fabricación informática integrada** (CIM).

Figura 1.1
Historia de JDF; logo CIP4

- 1993 Desarrollo del concepto de PPF
- 1995 Presentación pública de PPF Versión 1.0; Fundación de la organización CIP3
- 1996 PPF Versión 2.0
- 1998 PPF Versión 3.0
- 2000 Presentación pública de JDF; Fundación del consorcio CIP4
- 2001 JDF Versión 1.0
- 2002 JDF Versión 1.1
- 2004 JDF Versión 1.3
- 2008 JDF Versión 1.4



Figura 1.2
Distribución de la
información PPF

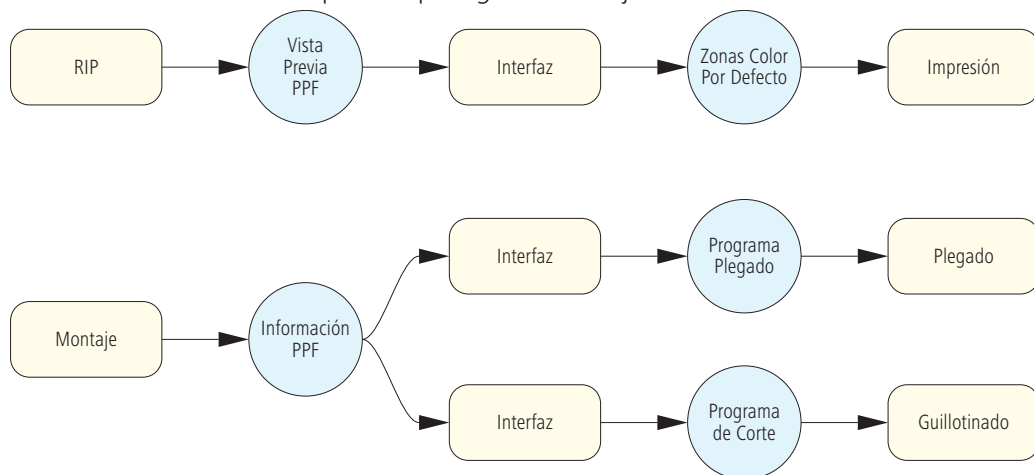


Igual que el “formato CIP3”, PPF es muy conocido en la industria gráfica y permite la transmisión de datos técnicos de preimpresión a prensa y a postimpresión (Figura 1.2). La aplicación más habitual del archivo PPF es la transmisión, esta puede ser de la imagen previa de una signatura (*Preview*) o de la tarea de un RIP completada y previa a la grabación de la plancha. Esta imagen se transfiere al programa para calcular los ajustes de tinteros de una prensa offset (Figura 1.3). Otro ejemplo es la transferencia de información para el corte o plegado de una hoja desde el diseño digital al acabado. La información se graba en el archivo PPF adecuadamente tras localizar las marcas correspondientes. El software de interpretación PPF, posteriormente, puede programar el corte en una guillotina o el doblado en la plegadora en el formato de datos patentado por el fabricante (lenguaje propio del proveedor del equipo). Por lo tanto, el PPF no transmite datos de control de máquina, transfiere la información abstracta de la cual se extraen los datos válidos.

En resumen, las ventajas de un flujo de trabajo PPF sobre otros métodos de producción son:

- Los departamentos pueden compartir los datos técnicos, especialmente para realizar los ajustes de máquina.
- Los datos no se insertan varias veces, como el tamaño de página.
- Al transferir los datos técnicos se pueden eliminar algunas etapas del proceso, por ejemplo el escaneado al inicio del proceso para generar los ajustes de color.

Figura 1.3
Flujo de trabajo PPF
detallado



- Con la normalización y publicación del **PPF (Formato de Producción de Impresión)** los módulos compatibles PPF de distintos fabricantes pueden comunicarse entre sí.

En la sección 4.2 encontrará más detalles sobre el Formato de Producción de Impresión PPF.

1.2 Características del Formato de Definición de Trabajo

En ocasiones, JDF se considera un “parte de trabajo electrónico” (ver figura 1.4), pero en realidad es mucho más. Se puede utilizar para controlar los flujos de trabajo automáticos, realizar preajustes de máquinas y registrar los datos del trabajo.

Lo más importante de JDF es que ofrece soporte a:

- La transmisión de datos del pedido.
- La transferencia de datos de configuración.
- La recopilación de datos de producción y máquina.
- La planificación de la producción.
- Seguimiento de los pedidos.

La funcionalidad de PPF está incluida en la especificación JDF pero los flujos de trabajo JDF/JMF incluyen opciones adicionales:

- Soporte del interfaz MIS entre los programas de presupuesto y producción (MIS sistema gestor de la información). JDF también ayuda a especificar la “información tipo software” del producto (por ejemplo, la frecuencia de trama estimada). Estos parámetros del proceso de producción deberán definirse en algún momento posterior.
- Datos de **seguimiento del trabajo** y cálculo de costes por orden (información de máquina y operaciones).
- Un protocolo para controlar los trabajos y equipos de producción.
- Funciones para definir “Objetos de Negocio”, como solicitud de presupuestos, ofertas, confirmaciones de pedidos, etc y dar soporte al comercio electrónico entre compradores y fabricantes de productos de impresión.

Figura 1.4
Anotaciones del parte de
trabajo en el transcurso de
la producción

Rechnungsnummer: 08-6017 v 4.808

Auftragsnummer: 0808-6771
 Alte Auftragsnr.: ~~0806-888~~
 Sachbearbeiter: _____

Termin: 11.11.08
 Druck: 30.7.
 Weiterverarb.: spat vorantrag
 Auftragsdatum: 27.08.2008
 Telefon: _____
 Telefax: _____
 Zuständig: 207
 Mobil: _____
 Best.-Nr.: Postkasten zustellen
 Abteilung: Reiter 11.7. entziffert

Neudruck Daten gestellt am: _____
 Nachdruck mit Änd. e-mail DVD/CD ISDN
 Nachdruck ohne Änd. Abholung
 Zuluhr
 Versand DPD / Post

ACHTUNG:

Rechnungsanschrift: _____

Fremd- arbeiten	Firma:	Tätigkeit:	Fällig am:
1000	1.000 Exemplare Produkt: Visitenkarten Format: 85x55 85 x 55 cm Druck: 1/1-farbig Silber / braun Material: Chromoluxkarton 250 g/m ² Verarbeitung: glatt beschneiden Verpackung: handlich in Kartons verpackt Versand: Lieferung frei Haus Stuttgart (1 Adresse)	(A) veränderte Lack- glanzeffekt	= 190,- Ein Ankl. 6N (A) 3N (B)
500	500 Exemplare Produkt: Visitenkarten nur Logo Format: 85x55 85 x 55 cm Druck: 1/0-farbig braun 1.877 C Material: Chromoluxkarton 250 g/m ² Verarbeitung: glatt beschneiden Verpackung: handlich in Kartons verpackt Versand: Lieferung frei Haus Stuttgart (1 Adresse)	(B) Dispersion- glanzlack silberfläche	= 85,- 35 x 25
21	1.000 Exemplare Produkt: Postkarten / Text 1 (Einkauf Neuauflage) Format: DIN A6 Druck: 1/1-farbig braun / Silber Pantone 877 C + Pantone 174 C Material: Bilderdruck matt 250 g/qm 30% recyceln Lage Verarbeitung: glatt beschneiden Verpackung: handlich in Kartons verpackt Versand: Lieferung frei Haus Stuttgart (1 Adresse)	1000	19.7. Einzahlungs- 185,- 165,- 35 x 25 26.8
31	1.000 Exemplare Produkt: Postkarten / Text 2 (neutral) Format: DIN A6 Druck: 1/1-farbig braun / Silber (+ Dispersionglanzlack Material: Bilderdruck matt 250 g/qm silberfläche) Verarbeitung: glatt beschneiden Verpackung: handlich in Kartons verpackt Versand: Lieferung frei Haus Stuttgart (1 Adresse)	1100	119,-
41	500 Exemplare Produkt: Briefbogen Format: DIN A4 Druck: 2/0-farbig braun, Silber Material: Offset weiß 80 g/m ² Verarbeitung: glatt beschneiden Verpackung: handlich in Kartons verpackt	600	152,- 2N
1. Lieferung frei Haus Stuttgart (1 Adresse) <u>Muskis Visiten</u> 5) Letz. Layout Gerüstung 315,544 2. <u>silber, braun in Farbe</u> Anzahl: <u>169</u> 3. <u>1300</u> Kartons: 2. <u>1</u> Versand: <u>100</u> Pakete: <u>100</u> 3. <u>1</u>			

- Establecimiento de un procedimiento inicial para integrar nuevos programas JDF/JMF en su entorno de producción mediante el método **enchufar y usar (plug and play)**.
- Creación de registros de producción para trabajos de impresión específicos.
- Métodos de extracción y transmisión de datos del JDF para el proveedor de planchas o para ser utilizados en otros programas o departamentos dentro de la imprenta. Las secciones JDF modificadas pueden ser reincorporadas al JDF original al finalizar su uso externo (figura 1.5).

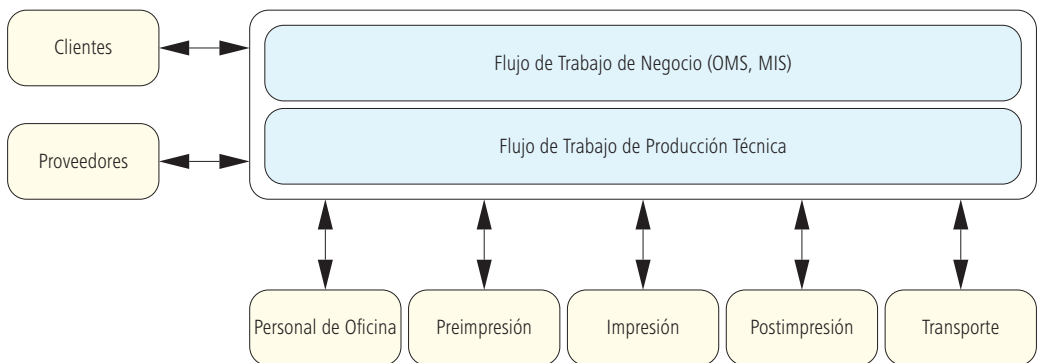


Retomaremos los detalles de estos puntos en próximos capítulos y entonces se entenderán completamente. Lo que podemos afirmar ya es que, si comparamos ambos flujos de trabajo JDF y JMF, JDF es mejor gestionando pedidos. (Figura1.6).

Pero ¿por qué es tan importante este aspecto?. La respuesta es muy simple: el menor tiempo de preparación de las máquinas de impresión y acabado permite procesar más trabajos por hora, reduciendo la media de tiempo por tirada y aumentando la tasa media de producción.

Figura 1.5 Modificación de un documento JDF cuando diversos proveedores de servicios participan en las tareas

Figura 1.6 Canales de comunicación en el flujo de trabajo JDF



Por lo tanto, la reducción más importante del coste, sin contar el tiempo empleado en la tirada, proviene de la rapidez en la gestión de pedidos y preimpresión.

1.3 Implementación de un Flujo de Trabajo JDF

JDF solo es el formato de datos, no es un flujo de trabajo. Cuando uno crea una aplicación de maquetación, la especificación PDF de Adobe no impone las reglas para que acepte este tipo de archivos. De igual modo, la especificación JDF no es una guía de diseño modular para los productores de flujos de trabajo.

Sin embargo, existen generalidades aplicables al flujo de trabajo JDF.

Los archivos no circulan libremente de un programa al siguiente. En teoría es posible, pero sería la pesadilla para programadores y usuarios. En su lugar, se centraliza la recepción de todos los JDF en una base de datos. Este centro neurálgico puede ser un sistema global de producción o un **sistema informatizado de gestión (MIS)**.

En la actualidad JDF/JMF no están presentes en todo el proceso productivo de impresión. Algunas máquinas de producción, especialmente en acabado, no se conectan a la red JDF/JMF, la información fluye mediante protocolos patentados o mensajes PPF. En otros casos, el tradicional flujo de trabajo CTP es el origen de flujos JDF más amplios. También encontramos conexiones entre el MIS e impresión. Sin embargo, la información canalizada a través de JDF/JMF es muy limitada. Por ejemplo, es inútil buscar operaciones JDF en los departamentos gráficos de una agencia de publicidad.

La especificación JDF define los contenidos susceptibles de intercambios entre los componentes del flujo de trabajo. No pregunta qué información debe estar disponible desde qué componente y quién debe analizarla posteriormente. Es como si los controladores de vuelo y pilotos intercambiaran los datos de altitud, velocidad o ritmo de descenso, pero luego nadie se responsabilizara del vuelo.

El consorcio CIP4 ofrece documentos específicos que tratan sobre las comunicaciones entre distintas clases de módulos [12]; su propósito es ampliar el flujo de trabajo JDF.

Las **Especificaciones de Conformidad e Interoperabilidad (ICS)** explican como comunicar el interfaz MIS al interfaz de impresión o el del MIS con el de preimpresión. En la sección 6.7 encontrará más detalles.

Transcurrieron varios años entre la publicación de la especificación JDF 1.0 en 2001 y la salida al mercado de las primeras aplicaciones compatibles con JDF. Desde entonces (2008) y con la actual especificación JDF en la versión 1.4, ya existen muchas aplicaciones compatibles con JDF. CIP4 publica regularmente un catálogo de servicios y aplicaciones JDF bajo el título “JDF Marketplace” [14]. En la edición 2012 del catálogo de Drupa se publicaron más de 100 productos y servicios (www.cip4.org/marketplace).

Además, todos los programas JDF pueden ser revisados bajo los requerimientos ICS y recibir una certificación. En el sitio web de la Organización CIP4 podemos encontrar un listado de los productos certificados (www.cip4.org/certification/certified_products.php).

Sin embargo, la mera existencia de las especificaciones ICS o la certificación no sugieren que podamos conectar indiscriminadamente, sin pruebas y a ciegas, los módulos de diferentes fabricantes. Las ICS proporcionan unas normas básicas insuficientes para ciertos productos impresos especiales. La complejidad está en los detalles y por esta razón, la creación o ampliación de una imprenta mediante la integración JDF siempre se debe considerar un proyecto. Es necesario aclarar con los fabricantes y usuarios todos los detalles con antelación y realizar las pruebas necesarias.

En el capítulo 13 encontrará una visión más amplia, con trucos, consejos y métodos de verificación para la implementación práctica de un posible proyecto [35].

Ejercicio:

Visite la página web oficial de la Asociación CIP4 (www.cip4.org). En particular lea la introducción de la especificación actual [13].

2 Conceptos Básicos del Flujo de Trabajo

En este capítulo, trataremos dos cuestiones completamente diferentes. En primer lugar, mostraremos la experiencia de tres imprentas que denominaremos X, Y y Z. La empresa X tiene un flujo de trabajo actualizado pero no integra características JDF. La empresa Y, sin embargo, ha conectado algunos componentes con JDF. La empresa Z, por último, está prácticamente conectada por una red JDF/JMF.

Tras esta vaga descripción, el tema se convierte en algo más abstracto. En la sección 2.4 describimos conceptos básicos, como *flujo de trabajo*, *sistemas gestores de flujos de trabajo* o los *partes de trabajo*. En la sección 2.5, las características comunes y concluiremos con el flujo de trabajo de una imprenta concreta.

2.1 Empresa sin Flujo de Trabajo JDF

La plantilla de la empresa X consta de 12 empleados. Años atrás, se instaló en el departamento de preimpresión un moderno **sistema gestor de flujos de trabajo** (WMS), sus componentes son una impresora de pruebas de contenido y un sistema de salida CTP. El WMS puede crear archivos PPF para el preajuste de los tinteros, pero la función nunca se ha utilizado. La planta dispone de 3 imprentas offset rotativas de 5, 4 y 2 colores y un formato común de 50x70 cm. También existe una imprenta digital con acabado, una guillotina, una plegadora y una máquina de gofrado y estampado de libros. Cada semana completan aproximadamente 50 trabajos de naturaleza muy variada.

Las solicitudes de pedido llegan por correo electrónico o teléfono y, en ocasiones, por fax. El personal de atención al cliente (CSR) introduce los detalles más importantes en el sistema de pedidos. El propietario suele realizar los presupuestos con lista de precios y calculadora. Aproximadamente un tercio de los presupuestos están incompletos y deben ser consultados por teléfono con el cliente.

En los últimos años, la revisión de presupuestos por parte de los clientes ha crecido de un modo espectacular. Las agencias de publicidad, en particular, solicitan la reducción del coste de sus trabajos.

El ratio de presupuestos aprobados y contratos representa menos del 10%. La mayor parte de datos referentes a los trabajos y contratos aprobados son enviados como datos adjuntos por correo electrónico.

El 80% son documentos PDF y el 20% restante son archivos In-Design, QuarkXPress o Microsoft Office. El servicio de atención al cliente (CSR) recibe y gestiona los documentos. A continuación se realiza la imposición, se asignan los trabajos a la prensa correspondiente y se imprime una orden de trabajo sin registrarla en el sistema gestor de órdenes (ver figura 1.6). Si el pedido está valorado en más de 500 euros se envía una confirmación al cliente.

El departamento de preimpresión de la empresa X examina todos los documentos con un programa de verificación previa (*PreFlight*), 2 de cada 3 archivos recibidos contienen errores. Si tienen solución en menos de 10 minutos se resuelven inmediatamente sin coste adicional ni consulta, se realiza una prueba digital de contenido y se envía al cliente por correo electrónico o transportista. Tras la firma y aprobación del mismo, se realizan las planchas. Solo se realiza una prueba de color en una empresa externa especializada si el cliente la solicita explícitamente. Si se cumplen los requisitos previos las planchas van a producción.

Las fechas de entrega se apuntan en un calendario de pared a modo de planificación. Por desgracia siempre no funciona porque atención al cliente olvida copiar algunos vencimientos en el calendario de la sala de prensa. A menudo, el cliente llama para cambiar la orden, atención al cliente busca el parte de trabajo e inserta los cambios.

Los procesos de reclamación (cobros pendientes) y facturación son compatibles con el sistema de gestión de pedidos. También existe un interfaz en línea que conecta con la administración tributaria.

Los cálculos de costes se realizan de forma manual y esporádica. Los costes diarios solo se calculan en ocasiones y se hace para verificar las listas de precios que utilizarán en presupuestos posteriores. El proceso para impresión digital es idéntico al de offset.

El propietario piensa que podría invertir en equipos automatizados de alta producción, pero la gestión de pedidos no alcanza los niveles de automatización necesarios para las máquinas y dispositivos de impresión.

Parece ser que es una idea extendida, sobre todo, en pequeñas imprentas.

2.2 Empresa con Integración Parcial de JDF

El trabajo normal de la imprenta Y son catálogos, revistas y encartes. Tiene 2 rotativas grandes. La mayor parte de portadas para sus productos se imprimen en la pequeña máquina offset tamaño 3B.

Las solicitudes de presupuesto proceden de clientes habituales y de los captados por los comerciales.

Los representantes disponen de un formulario en papel donde rellenan los datos del presupuesto, como número de páginas, formato, tipo de papel, colores, etc. Las órdenes se registran en el sistema de gestión de pedidos (OMS) junto a la información solicitada a los nuevos clientes. También se realizan y envían los presupuestos a los compradores potenciales. El personal de ventas realiza el seguimiento, renegociando y modificando el presupuesto si fuera necesario.

Al realizar el pedido, se imprime la información para el departamento de control y planificación de la producción (capacidad de prensa necesaria, fecha de pedido y entrega), quedando fuera del control del sistema de gestión de pedidos. A partir de entonces, los empleados de planificación gestionan el pedido insertando una tarjeta en el tablero interno del departamento. Estos datos no vuelven al sistema gestor de pedidos y si se producen incidencias tampoco las comunican de forma automática al sistema, lo que provoca conflictos ocasionales.

Un miembro del personal realiza el pedido del papel enviando una orden automática al proveedor tras consultar con el encargado de compras. El sistema envía una confirmación simultánea al cliente. El plan de trabajo (plegado, parte de máquina y paginación) se especifica en el sistema gestor de pedidos. También se introducen detalles de producción innecesarios para el presupuesto, como las especificaciones del RIP. Entonces, el sistema pone la información a disposición de la empresa generando un parte de trabajo electrónico en formato HTML. Preimpresión recibe un documento impreso por separado donde se enumeran todos los procesos; el personal debe firmarlo al terminar su labor. La información del sistema de flujo de trabajo de preimpresión se transmite por JDF, así muchos de los valores añadidos pueden ser transferidos automáticamente al sistema gestor de órdenes. Los datos con los preajustes de zonas de entintado (sección 4.2) se transmiten a las prensas en formato PPF para obtener el color correspondiente. También se envía una previsualización a la pantalla calibrada del panel de control.

A través del portal de internet, el cliente puede ratificar los productos menos complicados y confirmar la producción en prensa mediante la clásica prueba de color. También puede solicitar correcciones o autorizar trabajos que desembocan directamente en el sistema de flujo de trabajo de preimpresión.

Los trabajadores pueden insertar en los terminales todos los datos necesarios, como informes de planta, horas de personal, costes, tiempos de arranque, etc. Las prensas transmiten su estado a un sistema automatizado de máquina en red sin conexión de retorno con el OMS, por lo tanto, los datos de referencia deben ser insertados manualmente. A continuación el sistema de gestión de órdenes calcula las variaciones entre lo presupuestado y el coste real.

Este ejemplo despeja 2 incógnitas, la primera es la mejora de la productividad al conectar el interfaz JDF correspondiente al sistema gestor de pedidos (OMS). En segundo lugar revela el dilema JDF; la coexistencia de distintos entornos gráficos con formatos patentados produce una implementación parcial, con inserciones manuales en el sistema de gestión de pedidos y planificación y aprobaciones de impresión en línea.

2.3 Empresa conectada en red mediante JDF/JMF

La empresa Z es una imprenta plenamente integrada que cuenta con 150 empleados y ofrece todos los servicios. Imprime libros, folletos, revistas y documentos de empresa, como tarjetas de visita. Los pedidos se reciben normalmente por fax y correo electrónico. Los presupuestos se realizan en el sistema informático de gestión. El MIS, conectado con los sistemas de los proveedores, actualiza el precio del papel cada noche de forma automática. El inventario se realiza a diario y las existencias de papel necesarias para las órdenes de trabajo se verifican por internet directamente con el proveedor mediante un interfaz con protocolo propio de mensajería XML.

Si el presupuesto se convierte en orden, se genera un parte de trabajo en HTML directamente desde la base de datos, a disposición de todo el personal de la empresa.

El MIS produce un JDF por pedido con todos los datos relevantes que se transmite al área de producción mediante el servidor del flujo.

Los trabajos de preimpresión y planificación se gestionan electrónicamente con JDF, desde donde se obtienen y modifican la imposición, la maqueta y otros datos importantes para el proceso.

Los clientes también interactúan a través de la tienda web en línea. El sistema recibe y procesa los datos. El resultado de la verificación se registra en el archivo asociado al JDF y se puede enviar al cliente.

De las 6 prensas, 4 reciben datos como tamaño de hoja, zonas de tinteros y colores, vía JDF/JMF. La información de órdenes y ajustes de dispositivos se transmiten al servidor de forma recíproca. Hay 2 prensas antiguas integradas con terminales sin conexión directa. Por su parte, el servidor del flujo de trabajo envía de vuelta algunos datos recibidos de preimpresión e impresión al sistema gestor de pedidos, como los consumos de planchas, datos específicos del trabajo o momentos clave, como el final de la tirada o el agotamiento de las planchas.

Las máquinas de postimpresión y acabado no están conectadas en red y los datos se reciben exclusivamente a través de terminales manuales de entrada de datos.

Después de finalizar la producción, el sistema imprime una orden de transporte. Sin embargo, no hay conexión directa con la empresa de logística porque los envíos se realizan con varias compañías, a menudo son pequeños transportistas locales sin conexión al sistema.

Incluso en esta empresa, a pesar de su alto grado de conectividad, muchas áreas no han sido integradas con JDF/JMF. El acabado es un ejemplo de sector con soluciones específicas de los proveedores pero sus largos ciclos de amortización ralentizan la adopción de las nuevas tecnologías.

2.4 Definiciones

Un flujo de trabajo es un “flujo de producción” y su gestor es un sistema MIS. En el “río de trabajo”, el agua del año 1555 hoy es una orden de trabajo (ver Figura 2.1). Es necesario precisar los términos para no perderse en la inmensidad de la definición.

Definamos el término **trabajo cooperativo informatizado** (CSCW) como un campo de investigación interdisciplinar en grupo, especializado en las distintas tecnologías de comunicación necesarias. En el documento [41] se plantea este campo con diferentes perspectivas incluyendo algunos ejemplos. Los sistemas de software en grupo o **Groupware** son soluciones de software y hardware para CSCW.

Existen muchos y diversos productos groupware. Un sistema basado en el correo electrónico es un gran ejemplo, pero también

están incluidas en esta categoría las conferencias, la planificación y la edición grupal mediante sistemas de informáticos.

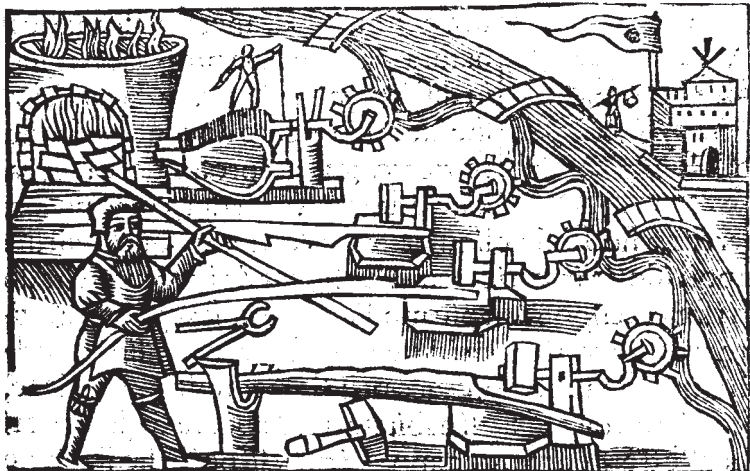
Entendemos por **flujo de trabajo**, la serie de pasos de un trabajo donde las secuencias de acciones se activan mediante eventos controlados y terminados. En principio, un flujo de trabajo puede ser cualquier trabajo secuenciado, no tiene porque ser por ordenador y un buen ejemplo sería cocinar. En la imprenta el trabajo se afronta de igual modo, los pasos son: preparación, cambio de planchas, ajustes, maculado, registro, prueba de color, definición de prueba de imprenta, control de tirada, etc.

La **gestión de flujo de trabajo** incluye la definición de las tareas o modelos, la gestión, ejecución y control y a veces la simulación de distintos flujos. Para seguir con el ejemplo anterior, la organización de una cocina o la programación de trabajos de imprenta, se pueden describir con una tabla de planificación del flujo.

Por último, el **trabajo cooperativo** que da soporte a la gestión del flujo de trabajo es un sistema **gestor de flujos de trabajo** (WMS). Por lo tanto, un WMS es la aplicación informática que ayuda a definir, gestionar y controlar la secuencia de las operaciones.

Si este tema le interesa especialmente, consulte el documento [31], en particular la página web de la Workflow Management Coalition (WFMC) [43], cuya definición de flujo de trabajo y sistemas de gestión de flujo de trabajo compartimos a continuación:

Figura 2.1
Olaus Magnus
Historia de gentibus
septentrionalibus (Historia
de los Pueblos del Norte)
de 1555
Forges de Dalécarlie
Bibliothèque
Sainte-Geneviève



Flujo de Trabajo: Automatización completa o parcial de un proceso de negocio, durante la cual se transfieren documentos, información o tareas basadas en normas de procedimiento desde un participante a otro.

Sistema de Gestión de Flujo de Trabajo: Programa informático que crea define y gestiona la ejecución de los procesos definidos sobre uno o más motores de flujos de trabajo con capacidad para interpretar e interactuar con los dispositivos, herramientas y aplicaciones informáticas necesarias.

Por lo tanto, hablar de **flujo de trabajo JDF** no es preciso. El término sería “Sistema Gestor de Flujo de Trabajo basado en JDF”. Pero como resulta un trabalenguas, lo denominaremos flujo de trabajo JDF a efectos prácticos.

El **Parte de Trabajo Electrónico (JobTicket)** es el contenedor electrónico de información utilizado por los gestores de flujo de trabajo para industrias gráficas. Por ejemplo, un parte de trabajo puede ejecutar una instrucción o un conjunto de parámetros para controlar la impresión. Estas órdenes electrónicas pueden describirse en Formato de Definición de Trabajo o en otras palabras, JDF es un ejemplo de parte de trabajo electrónico.

Un Gestor de Flujo de Trabajo o Worklow Management System (de ahí su acrónimo WMS) supera las funciones de los denominados **procesadores de partes de trabajo electrónico** o simplemente **motores de flujo de trabajo**. Estos módulos de software son capaces de producir, interpretar y ejecutar las funciones contenidas en los partes de trabajo electrónicos (como los ajustes de una prensa offset, el control de reventado o las conversiones de color).

Dichos partes proporcionan a preimpresión los denominados **metadatos** (información sobre otros datos). Los metadatos están vinculados y contrastan con los *datos de contenido* al mismo tiempo. Estos últimos son datos de impresión abiertos, como los procedentes de InDesign, Quark, etc, o formatos de intercambio cerrados como los PDF. Los metadatos se componen de “descripciones de objetos” e “instrucciones”. Una **descripción de objeto** es una imagen con 400 ppi de resolución por ejemplo y reducir esa imagen a 300 ppi es una **instrucción**. Los datos de contenido y los metadatos puede ser un solo archivo o estar separados como en un JDF (ver XMP, sección 4.1). Si los metadatos son externos debe existir una referencia a los datos de contenido, como el nombre del archivo.

Frecuentemente la frontera entre metadatos y datos de contenido es borrosa. Existe información de metadatos incluso en el nombre del archivo y con toda seguridad el WMS la utilizará para controlar el flujo de trabajo.

2.5 Clasificación de los Flujos de Trabajo

Los **sistema de gestión de flujo de trabajo** utilizados por la industria gráfica se dividen en 3 clases:

- Gestión de flujo de trabajo de departamento.
- Gestión de flujo de trabajo interdepartamental.
- Gestión de flujo de trabajo interempresarial con portal de enlace a clientes/proveedores en línea.

Estos sistemas se dividen a su vez en:

- Gestión de flujo de trabajo con conexión MIS.
- Gestión de flujo de trabajo sin conexión MIS.

La conectividad mediante el MIS juega un papel crucial y se denomina **integración del flujo de trabajo**.

Las categorías no difieren demasiado de la realidad. El sistema gestor del flujo de trabajo conecta segmentos de la actividad y algunas tareas no son compatibles con el WMS, incluso existen departamentos enteros sin posibilidad de ser integrados en el WMS interdepartamental. Por lo tanto, la cuestión está en el alcance del WMS. Un RIP, que a su vez proporciona datos a la prensa para ajustar el entintado es un ejemplo de WMS interdepartamental, también lo es una imprenta que ha integrado JDF en distintos departamentos.

A continuación explicamos las 3 categorías con más detalle.

La figura 2.2 muestra un esquema de flujo de trabajo tradicional. El cliente o persona que realiza el pedido comunica al administrador de la imprenta datos de contenido (flecha negra) y metadatos (flecha verde). Los metadatos se transmiten por teléfono o correo electrónico incluso, ocasionalmente llegan en CD o DVD. Tras abrir una orden de trabajo en el sistema gestor (OMS), el servicio de atención al cliente imprime un parte de trabajo y envía los datos a producción para validarlos y procesarlos. En el diagrama de la página siguiente vemos "Montaje" y "RIP" como ejemplo de los módulos de software responsables de procesar los datos.

Si hay más ordenadores o personas involucradas en el proceso de producción los datos se almacenan en un servidor de archivos, pero el parte de trabajo circula manualmente.

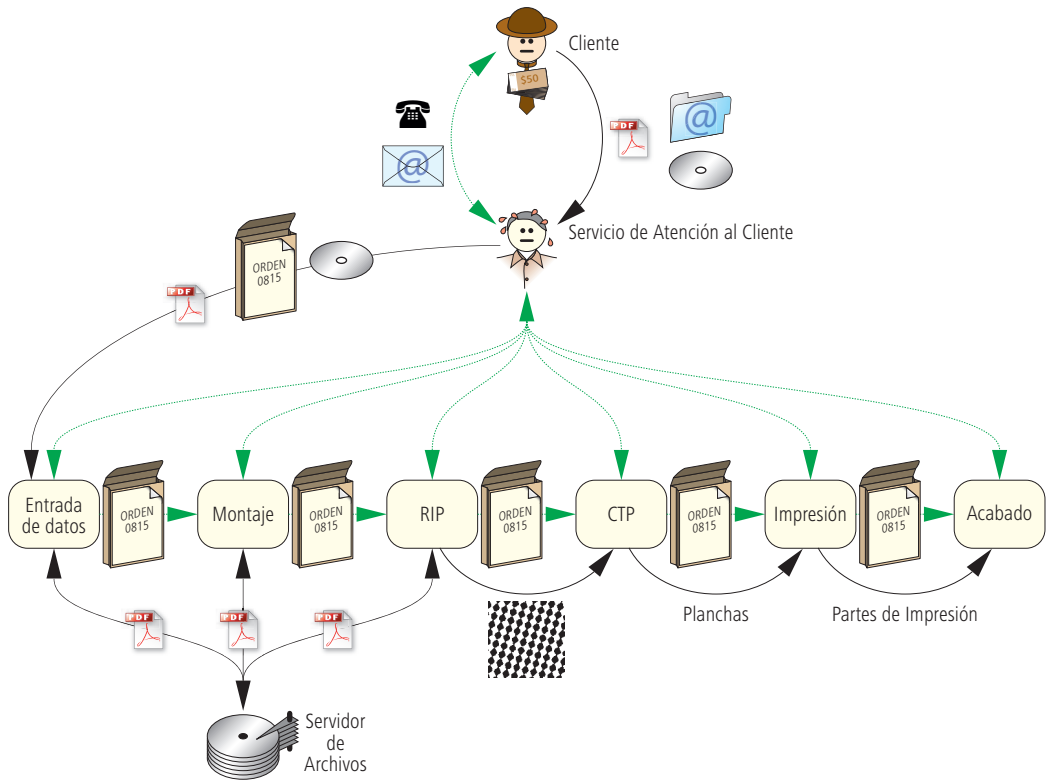


Figura 2.2
Flujo de trabajo de
Preimpresión tradicional

Finalmente se graba una plancha y se envía a prensa junto al parte de trabajo. El administrador debe contactar en persona o por teléfono con los distintos puntos del proceso de producción para el seguimiento del pedido.

¿Cuáles son los puntos débiles del flujo de trabajo?

Como mínimo debemos mencionar los siguientes:

- El pedido y los datos de contenido se comunican por varios canales reduciendo la fiabilidad del proceso.
- No se recopilan datos y si se hace es independiente del proceso de trabajo.
- EL seguimiento del pedido resulta muy caro.
- Siempre se hace el preajuste de máquina desde cero.

Este método no respeta la definición básica del sistema de gestor de flujo de trabajo.

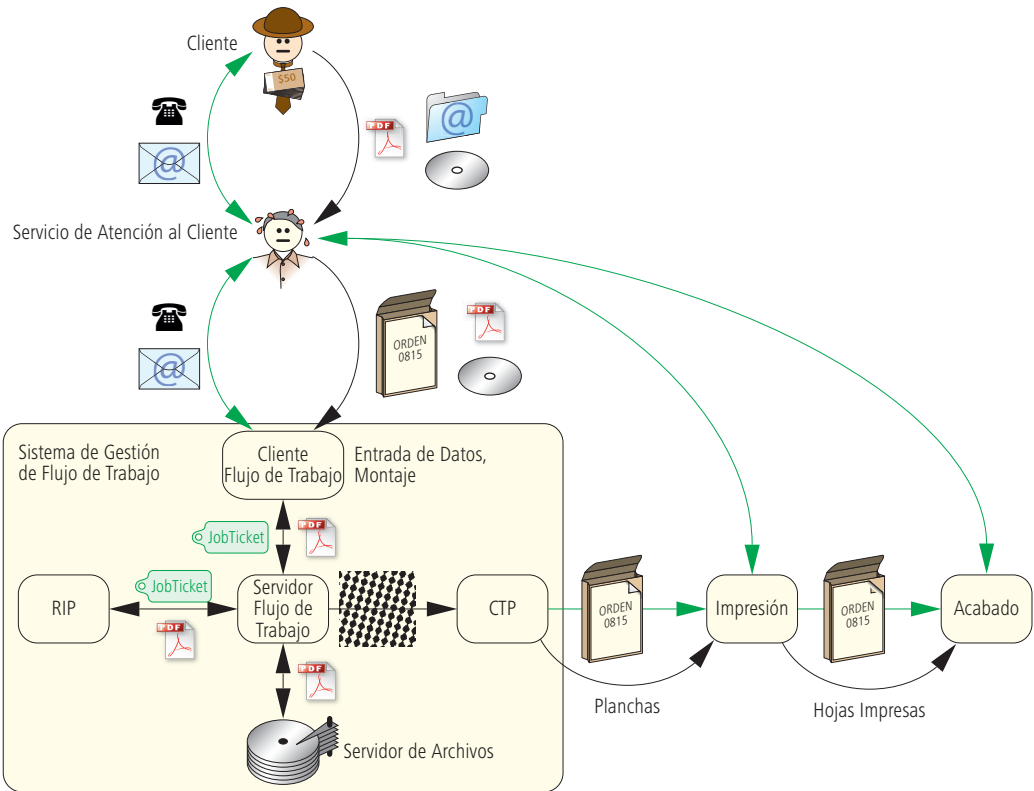


Figura 2.3
Sistema de Gestión
de Flujo de Trabajo
basado en el cliente

La figura 2.3 expone una gestión de flujo de trabajo basada en el cliente. Con el sistema cliente/servidor, la automatización de la producción resulta más eficiente.

El usuario accede como cliente a los partes de trabajo depositados en el servidor de flujo de trabajo. Habitualmente, el administrador del sistema es el único que puede cambiar datos por defecto de los partes de trabajo, por ejemplo, cambiar la lineatura del RIP a 70 líneas. Los operadores de preimpresión solo pueden cambiar los datos correspondientes a sus órdenes.

Al estar definidos de un modo estricto, estos partes de trabajo otorgan un alto grado de fiabilidad y flexibilidad al sistema, aunque si es necesario pueden ser personalizados.

Las fechas negras sobre fondo amarillo del servidor de flujo de trabajo transmiten datos de contenido y metadatos. En este caso, ha desaparecido la estricta separación entre los dos canales de comunicación que comentábamos antes.

Este concepto supone una importante mejora sobre flujos de trabajo basados en componentes aislados e independientes, aunque todavía tiene las siguientes limitaciones:

- No existe gestión general de pedidos ni de producción.
- No hay recepción de datos globales ni informes de rentabilidad.
- No se transfieren preajustes de máquina entre departamentos.

En el sistema WMS interdepartamental lo normal es la transmisión multifuncional de información electrónica, por ejemplo, preimpresión transmite a impresión los datos para calcular los preajustes de tinteros.

En la figura 2.4 se representa una empresa plenamente integrada mediante un sistema de flujo de trabajo interdepartamental.

Los metadatos se transfieren por red local a los distintos departamentos. El cliente gestiona los datos de contenido hasta que los recibe el departamento de preimpresión. Debemos tener en cuenta que la red está parcialmente instalada en la empresa, por lo tanto el plan no es completamente real.

En la figura 2.5, el intercambio de datos finales se produce íntegramente por internet. La gestión de presupuestos, pedidos, preimpresión y grabado de planchas puede ser realizada por distintas empresas ubicadas en cualquier lugar del mundo. La proximidad geográfica solo es necesaria cuando se manejan elementos físicos, como planchas u hojas impresas.

Aunque pueda parecer exagerado si consultamos ejemplos por internet basados en flujos de trabajo cliente/imprenta, podemos asegurar que no es tan disparatado.

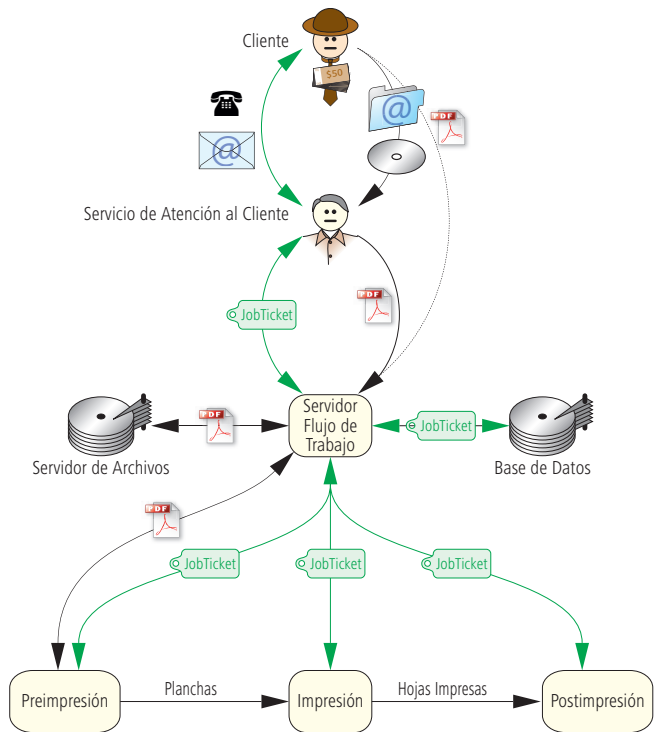


Figura 2.4 Sistema WMS integrado.

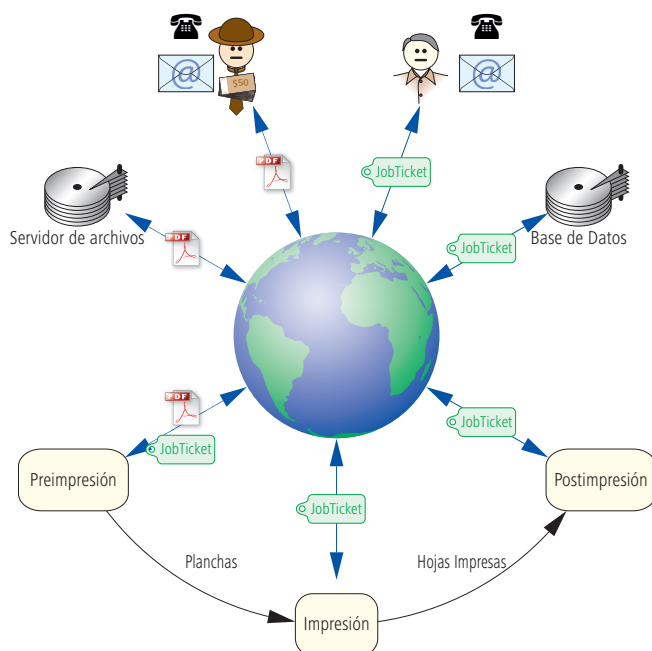


Figura 2.5
WMS basado en internet
(La ubicación de los
"Proveedores de servicios"
carece de importancia)

2.6 El Formato JobTicket y las Propiedades del WMS

Vamos a profundizar en los 3 requerimientos básicos de los sistemas gestores de flujos, concretamente en los necesarios para los formatos de ticket de trabajo subyacentes: adaptabilidad, normalización y capacidad de ampliación o extensibilidad.

Estas categorías pueden parecer bastante explícitas a primera vista, pero deben ser explicadas en profundidad.

Adaptabilidad

Los modelos de gestión deben adaptarse al flujo de trabajo del negocio y no al contrario. Esta afirmación precisa en apariencia, tiene sus limitaciones; no se puede adoptar un flujo de trabajo caótico. El sistema de flujo de trabajo se puede establecer de manera arbitraria pero el modelo de producción debe estar plenamente definido. De hecho, la tarea más importante y compleja a la hora de integrar un flujo de trabajo JDF en una empresa es analizar y definir el método de producción.

El orden habitual de los procedimientos comerciales y productivos es cronológico. Funcionan tanto en serie como en paralelo y también pueden ser repetitivos, recurrentes o superpuestos (figura 2.6). En la producción de impresos existen muchos ejemplos de las distintas situaciones:

- **Serie:** Primero imprimir, luego encuadernar...
- **Paralelo:** Escribir texto y editar imágenes para maquetar.
- **Superpuesto:** Planchas (diversos tipos), exponer e imprimir.
- **Alternativo:** Imprimir el trabajo en la máquina A o máquina B.
- **Recurrente:** Prueba, corrección, prueba, aprobación.

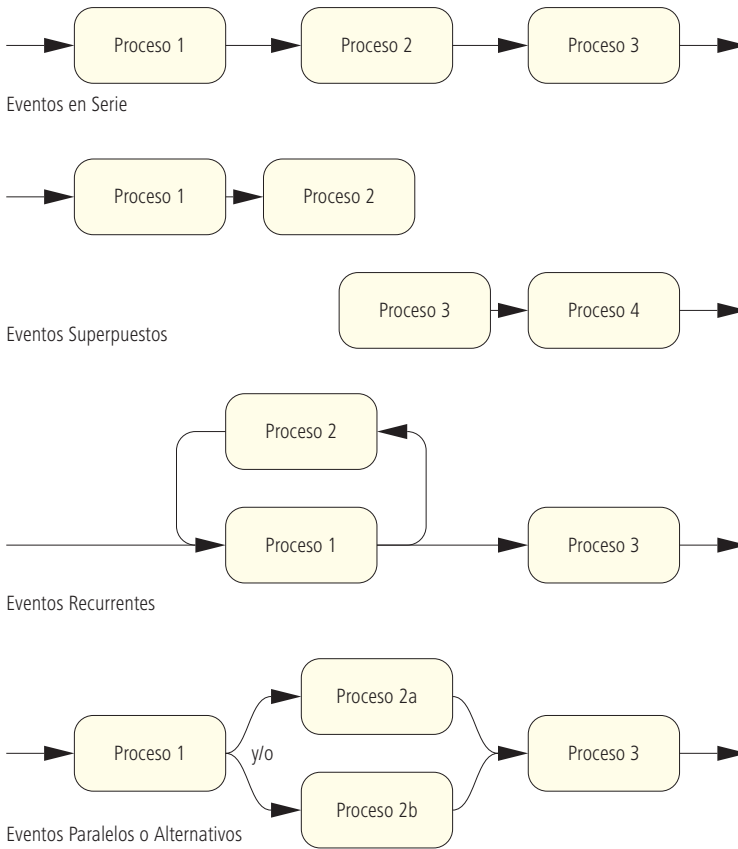


Figura 2.6 Distintas posibilidades de secuencia en procesos de producción y negocio.

Un WMS debe gestionar todas las posibilidades. La clave es diferenciar las funciones integradas en el WMS de las asumidas por el formato de archivo JobTicket.

Para comprender la superposición de procedimientos en JDF, se utiliza el concepto “tubería”. Cada operación genera una salida, (por ejemplo, la grabación de planchas) y esta salida es la entrada de un segundo proceso (impresión). Concretando, la salida virtual del primer paso es la entrada del siguiente (figura 2.7). El CTP va almacenando planchas en la tubería mientras se imprime. La estructura es como un almacén o silo .

Durante la producción de un impreso se actualizan detalles técnicos y organizativos. Por tanto, los partes de trabajo deben ser dinámicos y admitir nuevos datos. Por ejemplo, al aceptar la orden de trabajo se desconocen los ajustes exactos de las zonas de entintado. Además, en los estados iniciales habrá detalles sin definir. Ya hemos comentado que quizás conozcamos la resolución aproximada de salida, que ya estableceremos más tarde, durante la producción.

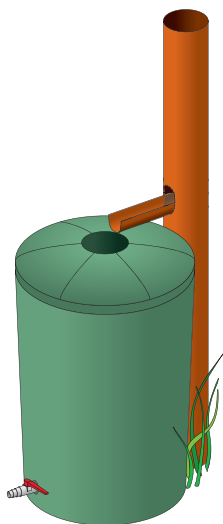


Figura 2.7
Cualquier jardinero aficionado conoce el principio de la tubería de agua de lluvia y el barril:
1) La salida de la lluvia es la entrada del barril.
2) Durante un chaparrón se puede llenar el barril por la tubería y regar las plantas al mismo tiempo.

Por lo tanto, el sistema de gestión y el formato de archivo deben permitir cambios durante el procesamiento del trabajo, incluso pueden ser habituales si así lo exige el cliente. Adaptar un sistema es muy complejo y exige un gran trabajo de desarrollo.

Normalización

Para integrar los **motores WMS** de distintos proveedores necesitamos normalizar la comunicación subyacente, aunque como en todo existen limitaciones. ¿Qué queremos decir con esta frase?

En primer lugar, significa que la orden de trabajo (JobTicket) debe ser completa en sí misma y contener la información necesaria para describir los productos gráficos. Si no fuera así, los proveedores no podrían usar el formato estándar por incompatibilidad con sus productos.

La alternativa es limitar sus funciones y definir qué se puede establecer como formato estándar. Pero imposibilitaría el progreso porque las empresas no podrían crear productos diferenciadores. Además estas normas son voluntarias, no son legislaciones nacionales y pretender restringirlo resulta totalmente irreal.

Las normativas sin plena aplicación están abocadas al fracaso. Por otra parte, los métodos de producción son muy complejos y la especificación de la norma nunca llega a completarse.

Entonces para resolver el dilema debemos admitir que la normalización llega hasta un punto y las órdenes de trabajo deben completarse con información sin normalizar. Este punto contradice la idea de la normalización y puede crear incompatibilidades entre los **motores del flujo de trabajo**, pero es la única manera de resolver la cuestión.

En breve trataremos la “capacidad de ampliación del formato” o extensibilidad del formato de archivo.

Para definir el contenido de las comunicaciones no basta con establecer “qué” hay dentro de la especificación de la orden de trabajo, también debemos establecer “quién” lo comunica.

Para que los distintos motores de flujos de trabajo se comuniquen correctamente deben identificarse claramente quien transmite la información y quien la recibe. Las **Especificaciones de Conformidad para la Interoperatividad** (ICS) de JDF/JMF son la orientación necesaria para este tema (comentadas en la sección 6.7).

Sin embargo, desde un punto de vista pragmático la normalización tiene una limitación. Los sistemas de flujo no se programan desde cero y van evolucionando con los años. A menudo, debemos tratar con formatos de archivo sin actualizar o sistemas JDF/JMF basados en viejas órdenes de trabajo o formatos específicos de algún proveedor.

Y haremos una observación relativamente trivial para finalizar: ¡Las normas evolucionan! Ya se han publicado 4 versiones de la especificación JDF y se plantea la cuestión de que los motores “hablen” y “entiendan” las diferentes versiones. (NdT: ya está disponible el borrador de la versión 1.5).

Extensibilidad o Capacidad de Ampliación

Aparte de cumplir con un modelo normalizado, un sistema WMS debe albergar espacio para añadir extensiones especiales. Esta demanda de ampliación es aplicable sobre todo al formato de órdenes de trabajo. Dado el requerimiento, ¿no debería integrar una función específica?

La estructura de la información contenida en la orden de trabajo se organiza según reglas y estructuras de otros idiomas (sintaxis). Estos idiomas son:

- PostScript (PS). Lenguaje de descripción de objetos.
- Portable Document Format (PDF). Documento portátil.
- Extensible Markup Language (XML). Lenguaje extensible.

El **formato de orden de trabajo PPF** (Print Production Format) y la **orden de trabajo portátil PJTF** (Portable Job Ticket Format) serán analizados en profundidad en el capítulo 4. Cabe destacar en este punto que PPF se codifica en PostScript y PJTF en PDF. El Formato de Definición de Trabajo JDF tiene estructura de documento XML. La extensibilidad de XML se explica en la Sección 5.2.

En la industria gráfica, los formatos PS y PDF ([4] y [5]) son conocidos como **lenguajes de descripción de página**, concretamente, como descripción de página y documento, respectivamente. Estos lenguajes se pueden ampliar libremente con claves propias o, para evitar conflictos de denominación, registrándolas en Adobe Systems Incorporated. Consideramos necesario explicar las posibilidades de expansión de PS mediante un ejemplo. El proceso para los PDF orientado a objetos es exactamente igual.

En PostScript, los nombres de variables se inician y definen con valores. El nombre de variable es una cadena y empieza con una barra inclinada (/). La palabra clave *def* asigna un valor a la variable. Por ejemplo, el siguiente código PS

/hev/Helvetica-Bold def

Define una variable con el nombre *hev* que, a su vez, contiene el valor *Helvetica-Bold*. En este caso, la variable sirve de abreviatura del nombre original. La siguiente línea se escribe de igual forma

/CIP3PreviewImageWidth 1425 def

El nombre de la variable *CIP3PreviewImageWidth* está definido en la especificación PPF y está normalizado que exprese el ancho de la vista previa en píxeles. De igual modo cualquiera podría definir sus propias variables con sus valores, por ejemplo:

/MiExtension/Sorprendente def

Este código PostScript es válido, pero no sirve para nada. Las empresas pueden transmitir en PPF información suplementaria para su flujo de trabajo con sus propias variables y demás estructuras PS, hecho que el PPF de la anterior organización CIP3 no permitía.

Las extensiones patentadas no describen “posibilidades teóricas”, actualmente se utilizan realmente en PPF, PJTF e incluso en JDF. Por esta razón, en determinadas circunstancias la compatibilidad entre sistemas se restringe. Las inserciones patentadas no deben reemplazar la información normalizada, deben complementarla. Los motores del flujo de trabajo ignoran las extensiones incomprensibles.

En resumen, aunque puedan parecer contradictorios debemos encontrar un punto en común entre *adaptabilidad*, *normalización* y *extensibilidad* para poder especificar el formato de la orden de trabajo de nuestro sistema de gestión de flujo de trabajo.

3 Modelos de Flujo de Trabajo de Impresión

En este capítulo, describimos 4 modelos de flujos de trabajo de impresión con la ayuda de algunos ejemplos:

- Lista de Actividades
- Diagramas de estado de transición y diagrama de actividades
- Diagramas de flujo
- Modelo productor-consumidor o proceso-recurso.

Pueden consultar otros métodos de procesos empresariales en [18].

Podemos describir las **listas de actividades** como lista de “tareas” y mostrarlo como la lista de la compra o un conjunto de recados. También puede incluir la lista de responsabilidades y las fechas de inicio y finalización del conjunto de tareas. Es una cuestión de quién, qué, cuándo y con qué recursos. Las consecuencias de las tareas parciales o los criterios para decidir las ramificaciones de dichos procesos no aparecen en estas listas. Este último aspecto es muy importante para los **diagramas de transición de estado**. Los estados en un diagrama de este tipo se dibujan como círculos o rectángulos y las transiciones son las flechas que los conectan. Un ejemplo son los estados de las máquinas, como *Estacionario*, *En Uso*, *En Espera* y *Error* (Figura 3.1). Por ejemplo, en caso de *Error* hay una transición de *Estacionario* a *En uso* o *Error*.

La descripción de flujos de trabajo utiliza el mismo tipo de diagramas, pero en lugar de estados representa actividades. Por lo tanto, las flechas representan la transición entre las actividades.

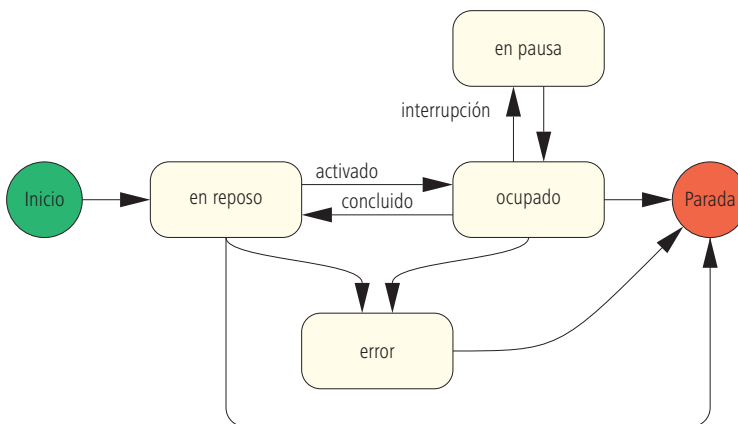


Figura 3.1
Estados de producción
de una máquina

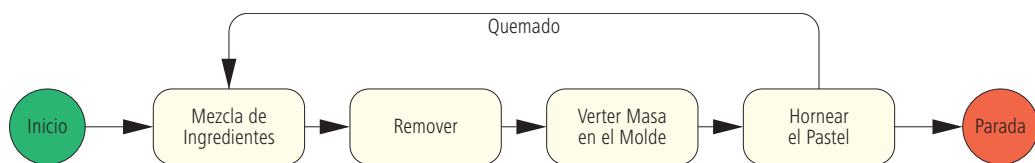
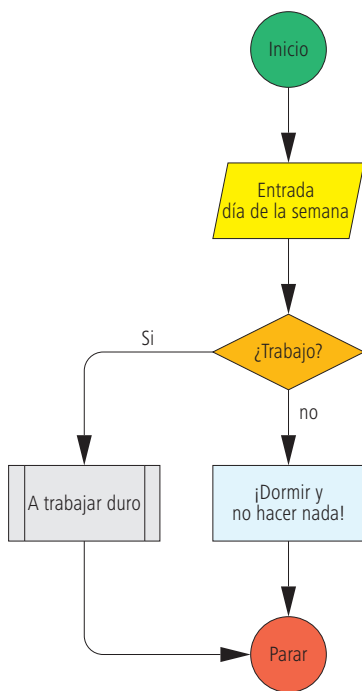


Figura 3.2
Diagrama de actividades
para hornear un pastel

Se pueden representar incluso recetas. (Figura 3.2). A lo largo de todo el libro los denominaremos **diagramas de actividades**.

Figura 3.3
Diagrama de flujo para
operaciones varias



Los **diagramas de flujo** sirven para visualizar las secuencias en el desarrollo de programas y aplicaciones. También se utilizan para describir procesos productivos o negocios, incluso pueden servir para planificar la vida diaria como vemos en la figura 3.3.

Los círculos verde y rojo, marcan el principio y final de un diagrama, respectivamente. El paralelogramo amarillo muestra la entrada o salida y el diamante naranja indica una opción que solo permite como respuesta si o no. Las operaciones más comunes están representadas por el rectángulo azul. El rectángulo gris con doble línea vertical es una subrutina que habitualmente se describe con más precisión en un nuevo diagrama.

El significado de estas formas geométricas está normalizado, pero los colores que les hemos asignado no.

El **modelo productor/consumidor** o **modelo proceso/recurso** describe lo que cabría esperar: un productor procesa un producto y lo coloca en un "servidor de datos" como si fuera la estantería de un supermercado. El consumidor toma un trozo de memoria y la utiliza. Ambos procesos están desconectados el uno del otro. Queda claro que primero el productor debe producir algo para que el consumidor pueda consumirlo.

Este modelo nos resultaría familiar en un flujo de impresión, si en lugar de productor/consumidor habláramos de proceso/recurso.

Por ejemplo, en producción de planchas un proceso graba el producto que consume el proceso de impresión. Impresión a su vez, genera la pila de hojas impresas, el recurso utilizado por el proceso de acabado y así sucesivamente (figura 3.4).

En el diagrama de transición de estados de la figura 3.2, esquematizamos el horneado de un pastel que también se puede describir perfectamente mediante un modelo productor/consumidor. Los procesos son la mezcla de ingredientes, remover, etc, los ingredientes son la harina, los huevos, la mantequilla y la leche; los recursos son los resultados de cada etapa, como la masa y el molde relleno.

Generalizando, podríamos decir que los procesos se corresponden con los verbos y los recursos con los sustantivos.

En el libro nos referimos a JDF como modelo proceso/recurso porque está basado en este principio.

Ahora aplicaremos estos tres modelos a las áreas de gestión de pedidos, preimpresión, impresión y postimpresión.

3.1 Gestión de Pedidos

Hasta la adjudicación de la orden las actividades de gestión de pedidos pueden simplificarse de la siguiente manera:

- Solicitud de presupuesto (cliente a impresor)
- Envío de presupuesto (impresor a cliente)
- Aceptación de la orden (cliente a impresor)
- Orden de confirmación (impresor a cliente)

La descripción mediante un diagrama de actividades como el de la figura 3.5 nos ayudará a ilustrar el proceso y la relación temporal existente entre las tareas parciales.

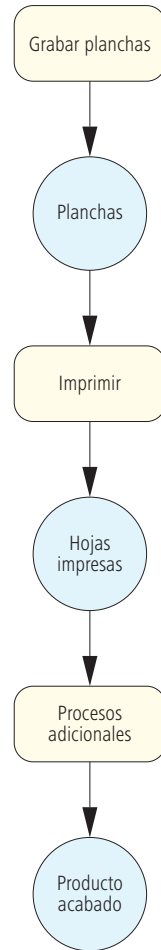


Figura 3.4
Flujo de impresión
productor/consumidor

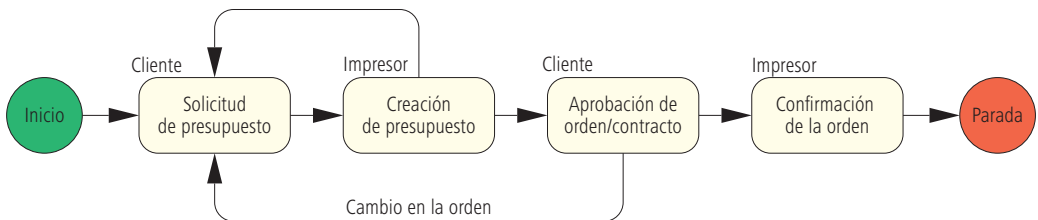
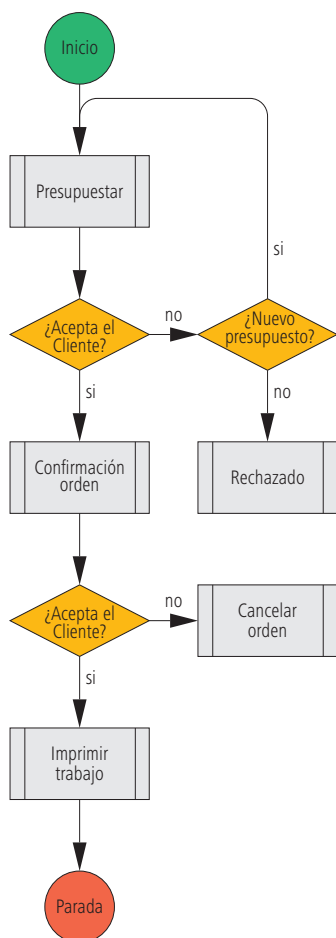


Figura 3.5
Actividades de la
gestión de pedidos

Figura 3.6
Diagrama de flujo con los
procesos de negociado de
una imprenta



En un flujo de trabajo cada transición puede generar nuevas operaciones que podemos ilustrar detalladamente con nuevos diagramas.

En la figura 3.6 ilustramos con un organigrama más detallado las tareas de imprenta descritas en la figura 3.5. Ver Figura 3 en [40].

3.2 Flujo de Trabajo de Salida en Preimpresión

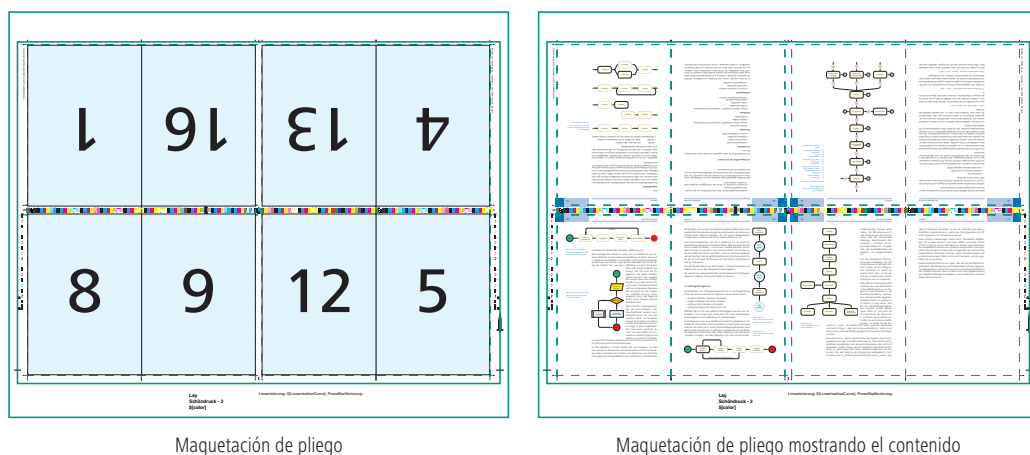
Cuando el departamento de preimpresión de una imprenta recibe un PDF con las páginas terminadas, las primeras tareas que suele realizar son las siguientes:

(Para aclarar términos consulte el glosario al final del libro).

- Preflight -- Verificación previa y corrección de datos opcional.
- Normalización de datos PDF.
- Conversiones de color.

A continuación prepara los datos de contenido para realizar las planchas propiamente dichas:

- Reventado y sobreimpresión (Trapping).
- Creación de maquetas de imposición (pliegos de prensa).
- Imposición.
- Rasterizado de datos para salida o prueba de contenido.
- Impresión de prueba de color para aprobación.
- Rasterizado de datos para imágenes de vista previa.
- Procesado de datos para generar la imagen previa.
- Procesado de datos para exponer y revelar planchas.
- Cálculo de valores para preajuste de las zonas de entintado.



Maquetación de pliego

Maquetación de pliego mostrando el contenido

Ahora veamos las diferencias entre “montaje” e “imposición”. En la práctica, estos dos conceptos no se distinguen con claridad. Lo primero que se define al crear la maqueta de impresión es el tamaño y los límites del papel (posición del pliego en la plancha). En las imprentas de bobina, el tamaño define la posición de las páginas y se disponen sobre un pliego de máquina virtual con una plantilla de imposición. Debemos contar con el tipo de impresión y la encuadernación respetando valores como el margen de pinza, los márgenes superior e inferior del folio y el área de desbarbe para la encuadernación adhesiva.

El orden de las páginas se define en plantillas sin contenido, denominadas páginas de muestra o marcadores de posición. Por último, se colocan todas las marcas en el pliego: corte, plegado, registro, barras de color y similares. El resultado es el diseño de imposición o pliego de impresión.

Para realizar la imposición, las marcas de impresión y las páginas contenidas habitualmente en un PDF se ordenan en una maqueta de imposición manual o automáticamente.

Como vemos en la figura 3.7, las plantillas de maqueta se rellenan insertando los datos de contenido en los pliegos. El resultado suele ser un nuevo archivo PDF.

Ambos procesos se realizaban anteriormente en el software de montaje o imposición. Hoy en día un operador trata las páginas y marcas de las hojas por separado en programas independientes y la imposición se procesa automáticamente en el servidor como tarea subyacente.

Figura 3.7
Pliego con maqueta
e imposición

En la figura 3.8, podemos observar el diagrama de actividades descrito con todos sus vínculos.

La descripción mediante el modelo proceso/recurso es más precisa, pero también resulta mucho más complicada (figura 3.9).

Los procesos están representados por rectángulos con cantos redondeados y los recursos con círculos, que podemos distinguir

en azul (desde R_1 a R_{10}) y rojo (desde R_{11} a R_{24}). Los recursos azules son detalles de ajuste para preparar los procesos correspondientes. Los recursos de procesado, por ejemplo R_7 , contienen datos del proceso de filmación (lineatura, tipo y ángulo de trama, etc.). El recurso de maqueta R_5 contiene información sobre el esquema de imposición, tipos de encuadernado e impresión, etc.

No hemos descrito los detalles contenidos en los recursos para ahorrar espacio, necesitaríamos un libro entero para mostrarlos.

La figura 3.9, enumera brevemente los recursos de los círculos rojos. El flujo de trabajo descrito en la figura 3.9 solo muestra la producción de planchas y su validez no se puede extrapolar a otros procesos. En el capítulo 12, veremos un procedimiento típico de la producción de envases completamente diferente.

Al nombrar los recursos en la figura 3.9, hemos hecho suposiciones que quizás no sean aplicables.

Elegimos la descripción "páginas PDF" para los recursos R_{11} y R_{13} ; sería más genérico, adecuado e incómodo hablar de "lenguaje de descripción de páginas secuenciadas".

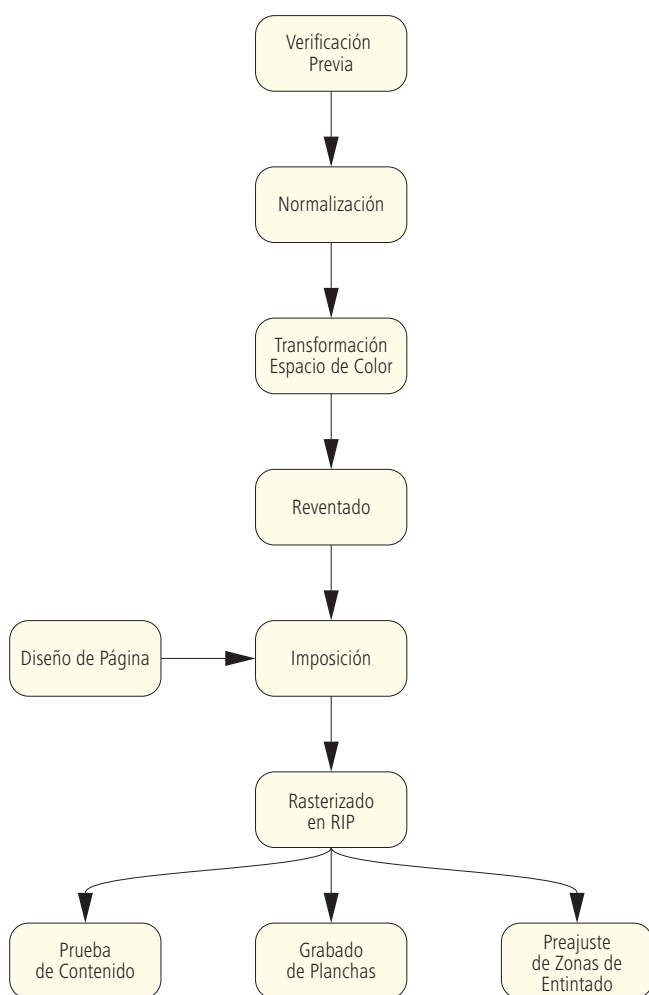


Figura 3.8
Diagrama de actividades
con sus vínculos

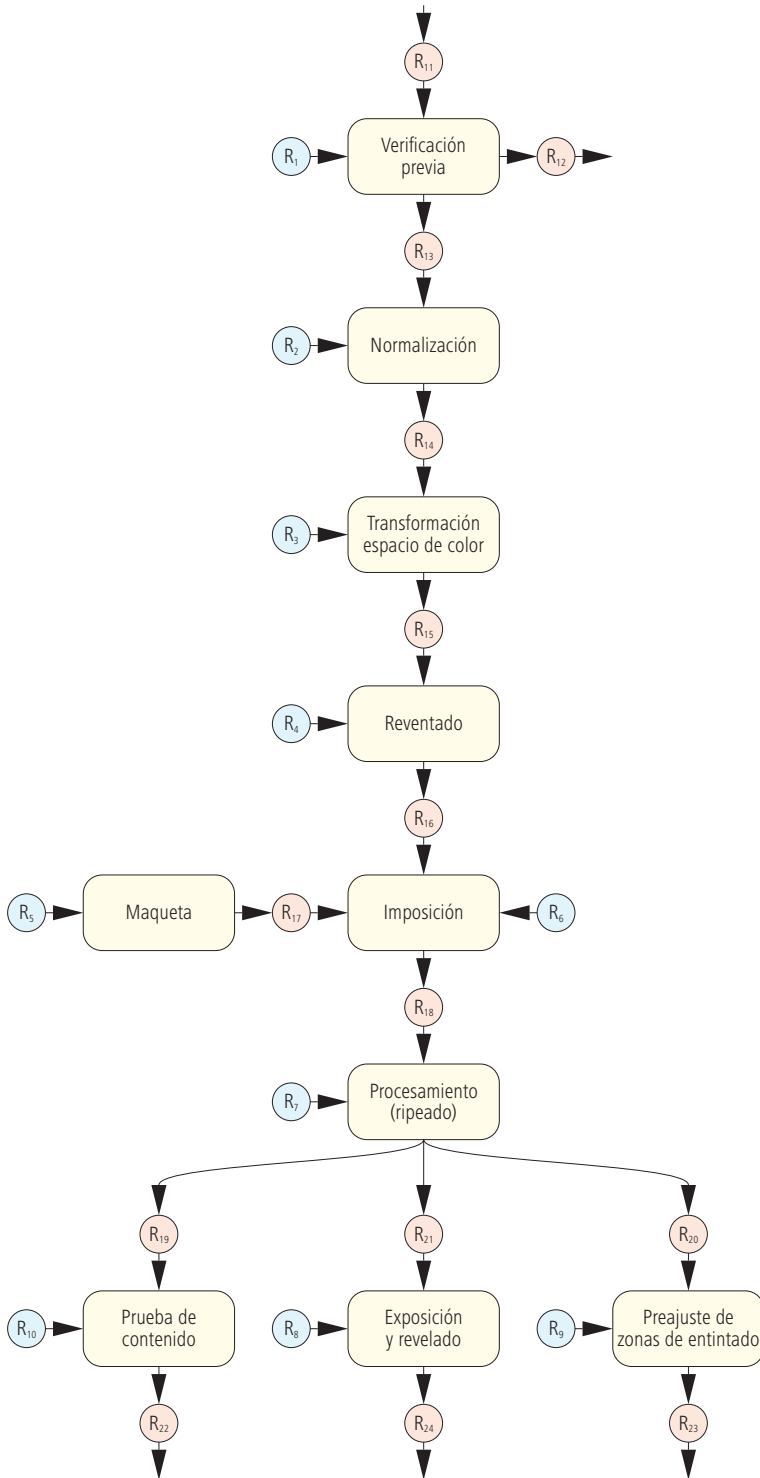


Figura 3.9
Modelo proceso/recurso
complejo.

Descripción de recursos:

- R₁ bis R₁₀ Parámetros de entrada de los procesos respectivos
- R₁₁ Páginas PDF
- R₁₂ Informe de verificación previa
- R₁₃ Páginas PDF
- R₁₄ PDFs normalizados
- R₁₅ PDFs conversión color
- R₁₆ PDFs con reventado sobre/infra aplicado
- R₁₇ Esquema de imposición
- R₁₈ Hoja PDF
- R₁₉ Archivo de una prueba de contenido
- R₂₀ Imagen Previa
- R₂₁ TIFF-B
- R₂₂ Prueba de contenido
- R₂₃ Valores de preajuste zonas de entintado
- R₂₄ Plancha de impresión con imagen

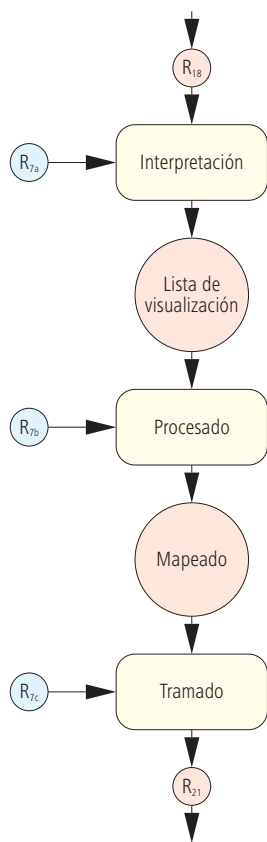


Figura 3.10
Proceso de rasterización

El recurso R_{11} es el resultado del proceso de maquetación. No son solo datos registrados, son recursos físicos y elementos con cualidades materiales. R_{22} es la prueba de contenido que se produce tras una consulta al cliente o va directamente al proceso de impresión. R_{24} es exposición y revelado y R_{23} el preajuste de zonas de entintado. Todos sirven como entradas a impresión.

En el modelo de la figura 3.9 se ven claramente las dependencias del modelo proceso/recurso, pero no describe la secuencia exacta. Por ejemplo no aclara si la prueba de contenidos se debe proporcionar primero y luego se producen las planchas o viceversa, naturalmente, esto último sería absurdo. En el capítulo 9 describimos los procesos de aprobación, dentro de la cadena de producción, que garantizan la correcta realización de las pruebas.

Llegados a este punto ¿cómo deberíamos estructurar el proceso?. Seguramente podríamos diseccionar el modelo de flujo con más precisión (como vimos en la figura 3.9) o definir unidades mayores (como las mostradas en la figura 3.4).

El proceso “pasado de planchas y revelado” con su plancha y exposición correspondientes, se puede dividir en: exponer, perforar, precalentar, revelado químico, lavar, engomar y hornear. Incluso, sería posible dividir el proceso de nuevo en: sacar la plancha del paquete, insertar en el tambor, fijar, exponer y extraer la plancha expuesta. Y así, podríamos estar detallando el proceso hasta el más mínimo detalle. Pero esto resultaría descabellado.

De cualquier modo, cada proceso de producción es un procedimiento ejecutado por un programa o dispositivo por definir o desencadenado por si mismo. También es verdad que nadie obtendría una plancha si no la fijara al tambor y la expusiera, por lo tanto dividir esta parte no tiene sentido.

Definir el proceso de planchas y revelado nos resultará útil ahora y nos puede servir en el futuro. Por el contrario, definir el proceso de la figura 3.4 no es factible porque los equipos y programas en producción de planchas y acabado son independientes.

En realidad, el modelo JDF es parecido al ilustrado en la figura 3.9. Sin embargo, hemos dividido el proceso de rasterización con más precisión.

En la figura 3.10 podemos reconocer los tres procesos: Interpretación, Procesado y Tramado. En consecuencia R_7 se ha dividido en tres recursos: R_{7a} *Parámetros de Interpretación*, R_{7b} *Parámetros de Procesado* y R_{7c} *Parámetros de Tramado*.

En la figura 3.11, explicamos brevemente los tres términos: El proceso de *Interpretación* analiza la estructura de los datos y la descripción de página. En *Parámetros de Interpretación*, por ejemplo, se puede indicar si la página se ajusta al tamaño de salida o debe reducirse. El proceso de *Interpretación* genera un conjunto de datos sin normalizar denominado *Lista de Visualización (DisplayList)*. El *Procesado* consiste en cerrar y rellenar los contornos definidos matemáticamente en el lenguaje de descripción de página, el resultado es una estructura de píxeles denominada *Mapa de Bits* y su resolución es la entrada de los *Parámetros de Procesado*. La estructura de píxeles obtenida está compuesta por diferentes tonos y tiene profundidad de color. El proceso de conversión de píxeles a mapas de bits puede producir tramas de semitonos AM o estocásticas FM. Los datos más importantes del proceso de tramado son entradas en los *Parámetros de Tramado* (tipo, frecuencia, etc). Pueden encontrar información detallada y ejemplos sobre estos procesos en [22] o en [29].

Pero ¿por qué dividir la rasterización en 3 procesos?. La razón es que en ocasiones, no es necesario el proceso de tramado completo, tal vez lo sea para obtener planchas o pruebas tramadas, pero no hace falta para una impresión digital.

Queremos recordar una vez más que no hablamos del propio flujo de trabajo, nos referimos al modelo de flujo de trabajo. Un recurso de salida contiene la descripción de las planchas expuestas, no las propias planchas, no obstante se refiere a un **recurso físico**, pero en esta publicación continuaremos hablando simplemente del “recurso plancha” y no vamos a diferenciar entre modelo y realidad.

En el modelo de flujo de trabajo no es necesario incluir los recursos de parámetros directamente. Por ejemplo, el archivo PDF que define los elementos de impresión de hojas o páginas no está incrustado en el JDF, se indica su localización en el sistema de datos, por lo tanto, únicamente se inserta la referencia al archivo PDF.

Tal vez se pregunte qué compone el modelo del flujo de trabajo realmente. El operador ¿debe preparar un flujo de trabajo automatizado por cada orden de impresión?. Si esto es así ¿debe configurar con exactitud todos los recursos? y si este fuera el caso ¿no es un gasto terrible?.

Bien, en la realidad esto no funciona de este modo. Pero entonces ¿de dónde obtiene el flujo de trabajo los datos para imprimir un trabajo?.

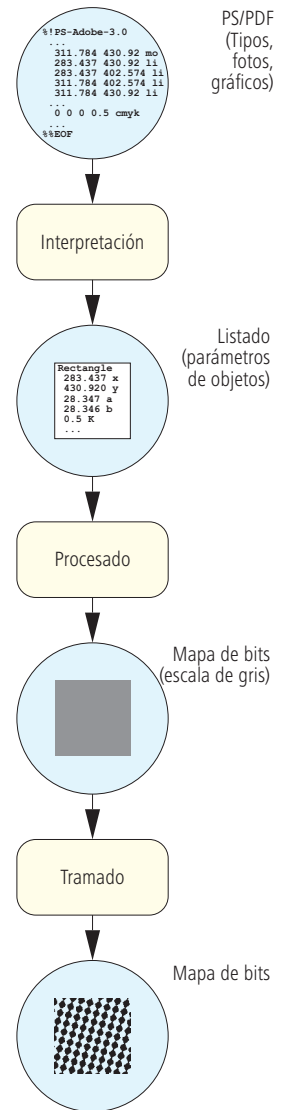


Figura 3.11
Proceso de rasterización y tramado

La respuesta es que se extraen automáticamente segmentos de información de las siguientes áreas:

- Sistema de gestión de pedidos/presupuestos.
- Datos de impresión procedentes del Cliente.
- Parámetros predefinidos de producción (valores por defecto).
- Formularios de datos gráficos en producción completados por los usuarios.
- Información de los trabajos de impresión aportados por el cliente mediante sistemas de comercio electrónico.

La descripción del flujo de trabajo se produce sin ayuda ni conocimiento del usuario en segundo plano. Los valores básicos y ajustes por defecto son suministrados por fabricantes, técnicos o administradores.

3.3 El Proceso de Impresión Offset de Hoja

Las lista de tareas y actividades habituales para imprimir un trabajo en una imprenta offset de hoja suelen ser las siguientes:

Tareas Básicas

- Leer la orden de trabajo.
- Preparar el papel.
- Realizar el ajuste de papel.
- Rellenar los tinteros con tinta.

Ajustes

- Cambio de planchas.
- Impresión, extracción y control visual del pliego o medición con dispositivo de color
- Ajuste de registro.
- Ajuste de color.

Ejecución de Producción (Tirada)

- Impresión, extracción y control visual del pliego o medición con dispositivo de color.
- Reajustes de color.
- Recarga de tinta.
- Cambio de la pila de papel.
- Lavar el cilindro de la mantilla.

Finalización de la Orden

- Lavar la mantilla del cilindro.
- Cambio de color, si es preciso.
- Completar orden de trabajo.

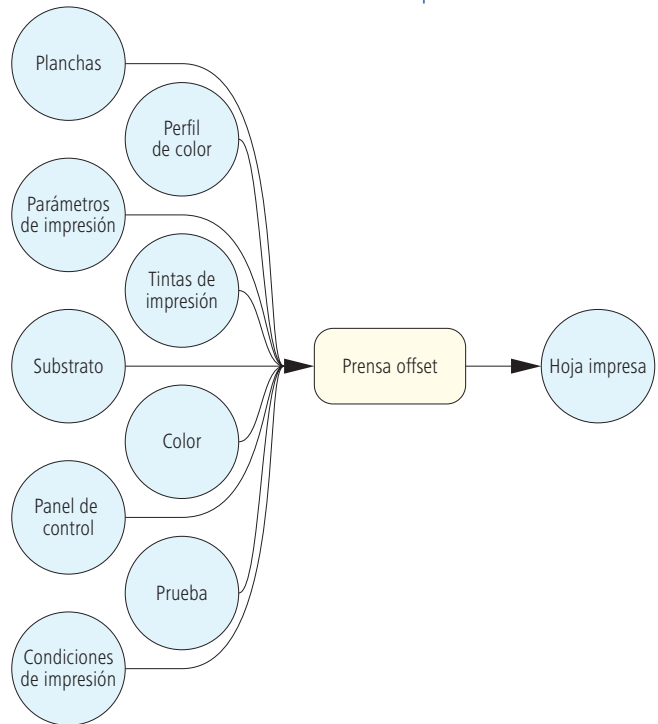
Entonces ¿necesitamos definir los 4 modelos: *Tareas Básicas*, *Ajustes*, *Ejecución de Producción* y *Finalización de Orden*? o ¿cada punto debe estar bajo su propio procedimiento?

Esto último no es una opción, porque las actividades dependen unas de otras. También resulta problemático dividirlo en 4 procesos porque los puntos subyacentes son muy similares o prácticamente idénticos.

En realidad, *Ajuste* y *Ejecución de Producción* son lo mismo, solo se diferencian en los resultados: uno es la maculatura y el otro las hojas correctas.

Debemos tener en cuenta que el orden de tareas es bastante estricto. Pero en la práctica el trabajo está “entrelazado”, es decir, durante la producción de un trabajo ya se está preparando el parte del siguiente.

Figura 3.12
Modelo JDF para impresión offset



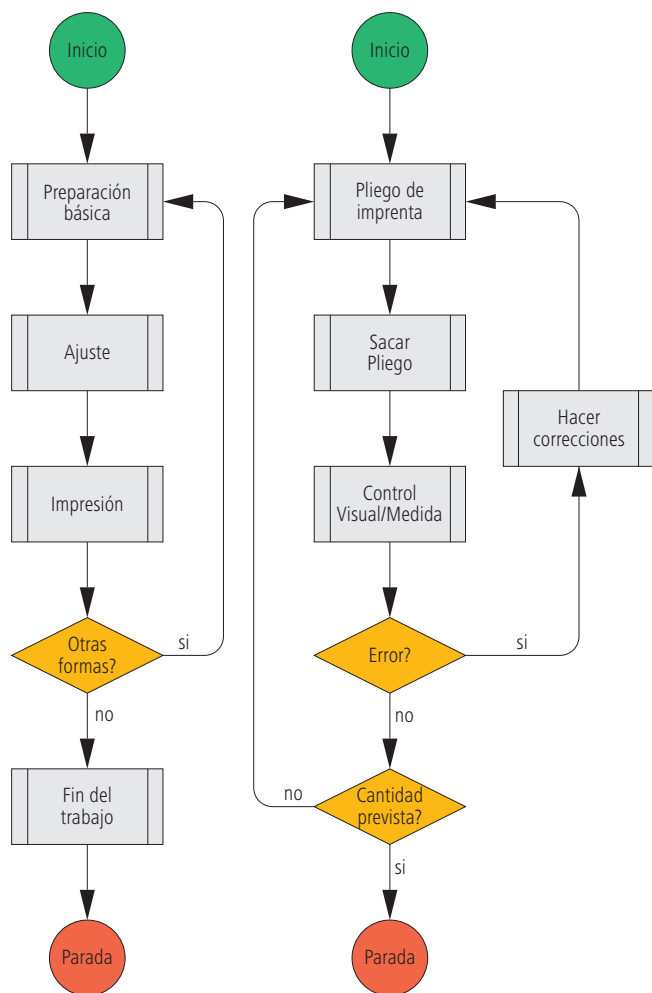


Figura 3.13
Diagrama de flujo para
impresión offset de hoja

En la parte izquierda de la figura 3.13, vemos el diagrama de flujo de las tareas principales mostradas como subrutinas. En el lado derecho está el detalle de la subrutina "Ejecución de Producción".

También encontramos subrutinas susceptibles de generar nuevos diagrama de flujo como "control visual/medición con dispositivo de color" y "extracción del pliego". De este modo se puede profundizar y llegar a niveles mayores de descripción.

Otro aspecto que debemos considerar es que ocasionalmente no es necesario cumplir todos los pasos. Por ejemplo, si el papel ya está en máquina "Realizar ajuste de papel" ya se cumple.

A este respecto, la Organización CIP4 ha elegido un modelo de proceso/recurso muy simple (ver figura 3.12). Para diferenciar las tareas en impresión offset hablamos de los "estados" del proceso.

Las imágenes, en impresión offset digital, se crean en la propia máquina. Si, además, dispone de acabado en línea, no solo graba las planchas e imprime, también realiza procesos simultáneos de corte, pliegado, alzado y grapado.

En la figura 3.12 podemos comprobar que el modelo es esencialmente teórico y algunos recursos son opcionales. Si desea un proceso más ilustrado puede crear su propio diagrama de flujo.

3.4 Modelo de Ejemplo para Postimpresión

El acabado de impresión es demasiado complejo para desarrollar un modelo que lo englobe en su totalidad. Realizar un listado universal de actividades podría ser demasiado largo y, aún así, estar incompleto, por tanto no tiene sentido intentarlo. En su lugar usaremos un folleto grapado como ejemplo para exponer un modelo de proceso/recurso.

- R₇ Pliegos.
- R₈ Signaturas.
- R₉ Hojas plegadas.
- R₁₀ Bloques recopilados.
- R₁₁ Bloques cosidos.
- R₁₂ Bloques recortados.
- R₁₃ Productos alzados y cosidos.

Los recursos de parámetros R₁ a R₆, contienen información sobre los procesos respectivos. A continuación, citamos algunas inserciones habituales para este tipo de recursos:

- R₁ Posición de bloques para el corte/marcas de corte.
- R₂ Secuencia, posición y sistema de plegado.
- R₃ Secuencia de alzado.
- R₄ Número, posición, ancho, ángulo, forma y perfil del cosido.
- R₅ Ancho y alto del producto final.
- R₆ Máxima altura y peso, número de capas por pila, número de impresos por capa

Generalmente, en R₁ aún no se ha insertado en el JDF el tamaño de la hoja de impresión, suele hacerse en R₇. Para el proceso de plegado las propiedades del papel carecen de importancia, por lo tanto no se insertan en R₂, se hace habitualmente en R₈.

Con este último, ya hemos visto ejemplos de flujos de trabajo descritos como diagramas de flujo y modelos de proceso/recurso para preimpresión, impresión y postimpresión.

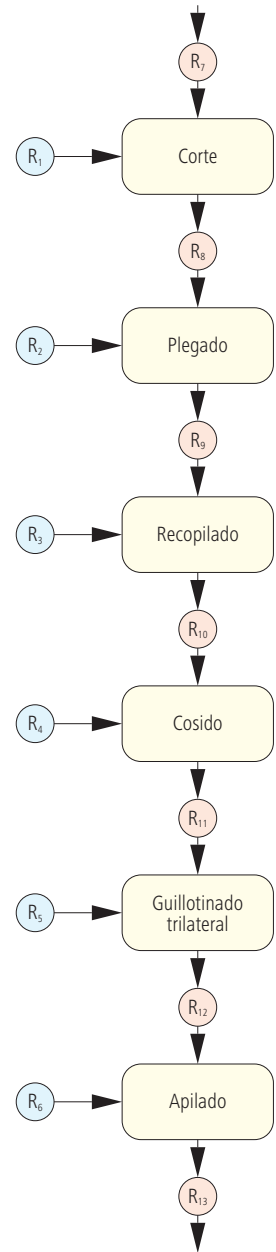


Figura 3.14
Modelo de Proceso/recurso para la producción de un folleto grapado

Consideramos este modelo especialmente adecuado para visualizar los procesos más complejos de forma clara y sencilla. En cambio, para ilustrar las “ramificaciones” dentro de un proceso, resulta más útil un diagrama de flujo. Quizás lo más sencillo es crear listas de tareas, pero al no ser muy descriptivas, pueden desembocar en un barullo de entradas sin estructurar resultando más parecido a una “tormenta de ideas” sin valor alguno.

Ejercicio:

Organice su próximo fin de semana con la ayuda de un diagrama de flujo, definiendo actividades alternativas si las condiciones no permiten cumplir alguna de las programadas.

Describa su plato favorito detalladamente con un proceso modelo/recurso, ¡sin olvidar un aperitivo y un postre!

4 Historia y Aplicación de los Metadatos

Ya definimos la diferencia entre *metadatos* y *datos de contenido* (datos para impresión) anteriormente en la sección 2.4. Ahora presentaremos las diferentes aplicaciones mediante algunos ejemplos.

En la sección 4.1 trataremos el formato EXIF, portador de datos técnicos y de calibración obtenidos al realizar fotografías o filmaciones con cámaras digitales. Además, presentamos el estándar de fotografía IPTC (International Press Telecommunications Council).

Extensible Metadata Platform XMP es la especificación de Adobe para formatos de imágenes y documentos. Bajo este concepto se esconde la idea de transmitir información sobre las limitaciones de uso y aplicación de fotografías.

Por ejemplo, los metadatos contenidos en un archivo de imagen TIFF se encuentran en la imagen colocada en el documento InDesign y en el PDF generado al exportar. Por lo tanto, la información de derechos de autor de las fotografías queda almacenada en el archivo PDF, incluso años más tarde.

El **Formato de Producción de Impresión PPF** tratado escuetamente en el capítulo 1 es un importante predecesor del Formato de Definición de Trabajo JDF y se explica en la sección 4.2 detalladamente.

La última sección de este capítulo trata sobre otro precursor de JDF, el **Formato de Orden de Trabajo Portátil PJTF**, desarrollado por Adobe Systems Incorporated. Este formato está estrechamente vinculado con la obsoleta **arquitectura Extreme RIP** de Adobe. Esta tecnología fue utilizada ampliamente por los fabricantes de sistemas RIP/CTP durante años y todavía se encuentra en el mercado.

El contenido de este capítulo no es necesario para comprender los demás capítulos explicativos de JDF y puede omitirse.

Muchas instalaciones JDF/JMF, PPF y PJTF actuales utilizan una gran diversidad lingüística de metadatos. En el libro trataremos únicamente los fundamentos de los metadatos más habituales.

Al margen de esto, la utilización conjunta de XMP y JDF es factible y su uso puede mejorar el rendimiento de la implementación del flujo de trabajo.

4.1 Metadatos para Fotos y Documentos

Existen 3 razones para incluir metadatos en una fotografía:

- Las fotos se identifican fácilmente en grandes bibliotecas de imágenes.
- Los derechos de autor contenidos en las fotos pueden evitar infracciones.
- El análisis de los datos técnicos y ajustes de cámara de las fotos, permite decidir la idoneidad para diferentes aplicaciones.

Lo más importante en el diseño de metadatos son: (1) la definición de la información que registramos y (2) como se conservan los datos cuando se convierte el formato.

EXIF y IPTC

EXIF es un formato de **archivo de imagen intercambiable** desarrollado especialmente para cámaras digitales por la (JEITA) [27] **Japan Electronics and Information Technology Industries Association**. Ciertos programas de escaneado admiten estos metadatos. Al disparar una fotografía, la cámara digital almacena información sobre si misma en el archivo resultante; son el instante exacto de la toma y los ajustes utilizados (resolución, apertura, tiempo de exposición, etc). Existen aplicaciones que pueden visualizar, analizar y modificar estos datos. En la figura 4.1 podemos ver los datos que proporciona automáticamente una cámara digital.

Aparte de los campos que muestra la figura, la versión 2.2 de la especificación EXIF [28] define muchos más, como los derechos de autor, comentarios o información GPS. También describe la estructura de datos EXIF para archivos de imagen JPEG y TIFF. Otros formatos no son compatibles, como PDF. Para transportar las entradas EXIF a PDF la información se estructura de otra forma. En la sección siguiente sobre XMP, explicamos el modo.

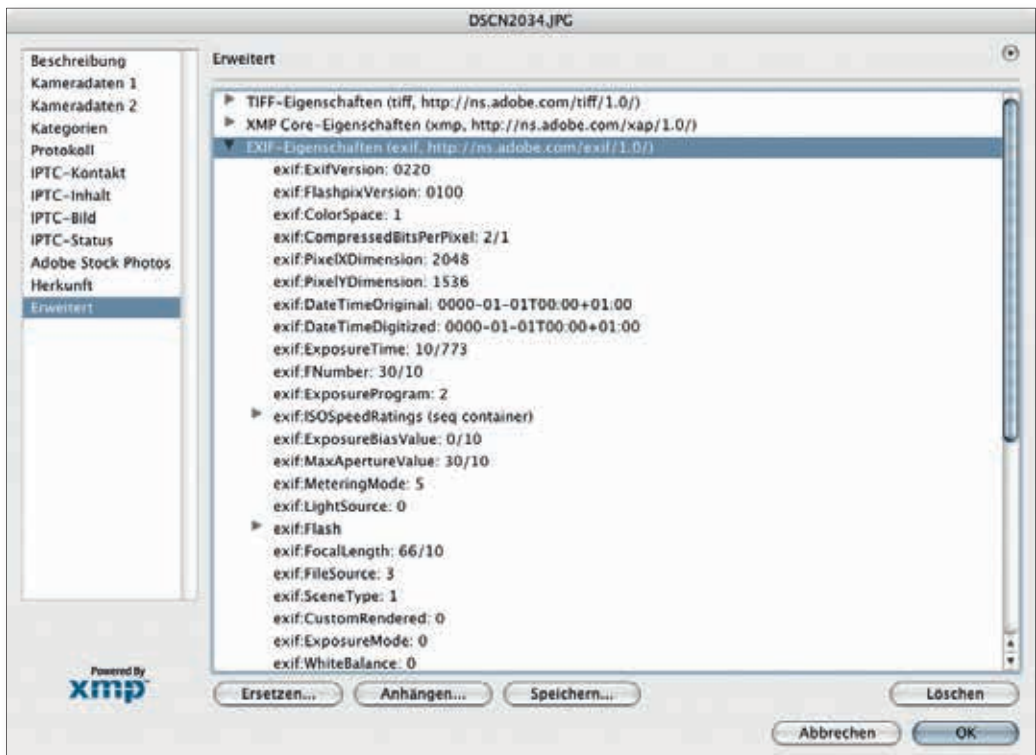
TIFF y JPEG son formatos extensibles y admiten la inserción de información EXIF. La primera sigla de TIFF "T" significa *Tag (Etiqueta o Etiquetado)*. Cada unidad de información de la imagen registrada (alto, ancho, etc) se graba con un valor definido en forma de número entero positivo. En total hay 215 etiquetas (=65536) y hasta ahora solo se han documentado unas pocas pero se pueden definir muchas más [3].

La extensibilidad del estándar JPEG utiliza una técnica similar. Normalmente las imágenes comprimidas se guardan según el **Formato de Intercambio de Archivos JPEG** (JFIF) [20]. Las unidades individuales de información se guardan con identificadores igual que en TIFF, pero se denominan “segmentos”, no “etiquetas.” El número de datos EXIF, por ejemplo, es 255.

Algunas cámaras digitales guardan en formato patentado RAW. En este caso, los datos EXIF se guardan en un archivo de texto separado.

El **Consejo Internacional de Telecomunicaciones de Prensa** [26] es una asociación mundial de asociaciones nacionales de prensa y agencias de noticias. En la década de los 90 estableció el **Modelo de Intercambio de Información** (IIM), es un conjunto de propiedades de metadatos aplicables a las imágenes similar a EXIF en muchos aspectos que contiene información sobre derechos de propiedad y se utiliza para describir las imágenes mediante palabras clave (Figura 4.2).

Figura 4.1
Datos EXIF de una
imagen JPEG



The screenshot shows a software window titled "DSCN2034.JPG" with a sidebar on the left containing the following menu items: Beschreibung, Kameradaten 1, Kameradaten 2, Kategorien, Protokoll, IPTC-Kontakt (highlighted), IPTC-Inhalt, IPTC-Bild, IPTC-Status, Adobe Stock Photos, Herkunft, and Erweitert. The main area is titled "IPTC-Kontakt" and contains the following fields:

- Ersteller: Carl Cool
- Berufsbezeichnung des Erstellers: Eisbär
- Adresse: Am Kaltenbach 3
- Ort: Eisleben
- Staat/Provinz: (empty)
- PLZ: 6295
- Land: Deutschland
- Telefonnr.: 09876/54321
- E-Mail-Adresse(n): carl.cool@gibb's.net
- Website(s): www.gibb's.net

At the bottom of the window, there is a "Powered By xmp" logo and a footer with the following text: "Mit Metadaten-Vorlagen können mehrere Felder gleichzeitig ausgefüllt werden. In der oberen rechten Ecke dieses Dialogfelds können Sie auf Metadaten-Vorlagen zugreifen. Support und Updates für diese IPTC-Fenster finden Sie unter <http://www.iptc.org/iptc4xmp>". The window also features "Abbrechen" and "OK" buttons.

Figura 4.2
Datos IPCT incluidos
en un archivo JPEG

XMP

En 2004, Adobe revisó la especificación **Plataforma de Metadatos Extensible (XMP)** que define los metadatos de imágenes utilizados para los distintos tipos de documento [2]. La idea es conservar los metadatos al cambiar de formato. Esto supone que las aplicaciones capaces de modificar metadatos puedan alterar datos XMP e incorporar metadatos en formatos especiales (TIFF, JPEG, JPEG 2000, GIF, PNG, HTML, PDF, AI, SVG/XML, PSD, PostScript, EPS).

De hecho, aunque esta no es la opción habitual, la información XMP se almacena separada de los archivos ya que la mayoría son entradas en bases de datos.

Los datos XMP pueden pertenecer a sus componentes individuales y al documento. Los gráficos e imágenes incrustados en un PDF están vinculados a sus propios datos XMP, como podemos ver en el ejemplo de la figura 4.3. La figura 4.4 muestra la información XMP y la página del libro. Un recurso puede ser un componente o el documento en sí y en lenguaje XMP se utiliza el término "recurso".

Los datos XMP pueden almacenarse en sintaxis XML. Este lenguaje es extensible mediante esquemas y se pueden utilizar varios simultáneamente. Hasta podemos crear nuestros propios esquemas. Hablaremos de XML en el siguiente capítulo. Adobe dispone de diferentes esquemas predefinidos, por ejemplo:

- Dublin Core Schema
- XMP Basic Schema
- XMP Rights Management Schema
- XMP Paged-Text Schema
- IXIF Schema for IXIF-specific Properties

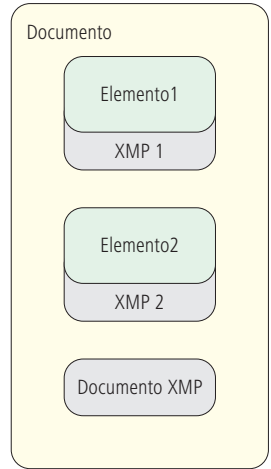


Figura 4.3
Componentes principales XMP de un documento



Figura 4.4
Información XMP de una imagen contenida en un archivo PDF



Puede consultar la lista completa en [2] aunque aquí solo enumeramos algunas entradas de las muchas posibles.

Dublin Core Schema describe las propiedades *Título* y *Lenguaje*. **XMP Basic Schema** captura el nombre de la aplicación que produjo el recurso (*CreatorTool*) y la fecha de creación (*CreateDate*). **XMP Rights Management Schema** alberga los derechos de propiedad. **XMP Paged-Text Schema** permite registrar los números de página (*Npages*) de un documento. La mayoría de entradas del esquema EXIF se definen en XMP. Esto permite integrar en estructuras XMP los metadatos de fotos digitales.

Los esquemas definidos por Adobe y otras compañías se pueden utilizar para insertar información propietaria del proveedor. De esta forma los fabricantes pueden utilizar las opciones XMP para transmitir metadatos y controlar su flujo de trabajo (WMS). En la

figura 4.5 vemos un ejemplo de lo que ofrece XMP. Describimos la inserción del software de un proveedor en un flujo de trabajo existente. En el paso 1 la aplicación envía los datos de la orden de trabajo en PDF al flujo de trabajo y guarda la ubicación de cada archivo en una entrada XMP. En el paso 2, se realiza el trabajo y como consecuencia se modifica el PDF. Por último, los datos se copian en el orden original del flujo de trabajo según las entradas XMP.

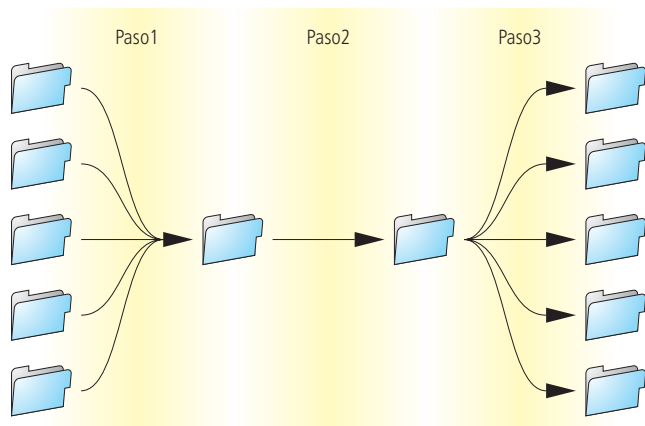


Figura 4.5
Software de terceras partes
en un flujo de trabajo de
preimpresión (WMS)

En resumen, los metadatos EXIF, IPCT y XMP definen las propiedades de los recursos digitales, es decir, son descriptivos. Las propiedades principales de XMP son limitadas pero pueden expandirse.

Este tipo de metadatos no describe los procesos necesarios para modificar los datos pero los programas de producción pueden utilizarlos para decidir entre alternativas de diseños predeterminados. Los metadatos se integran con los datos del archivo, por lo tanto están limitados al área de preimpresión que es la que gestiona los documentos digitales.

A primera vista, los formatos de metadatos que trataremos en las próximas secciones son completamente diferentes.

4.2 Formato de Impresión de Producción (PPF)

En la introducción hemos presentado las 2 aplicaciones más importantes de PPF: la transferencia de datos de preimpresión a prensa offset para calcular las zonas de entintado y los ajustes de corte y plegado para postimpresión (acabado). También comentamos en un breve ejemplo de la sección 2.6 "extensibilidad" que PPF está codificado en PostScript.

Figura 4.6 Estructura de archivo PPF

En la figura 4.7 mostramos la descripción del pliego donde la estructura lógica de la información PPF se construye como un árbol.

La hoja es el **elemento raíz** y las ramas del árbol son las bifurcaciones. Todos los puntos de información sin ramificación son hojas. En lugar de hojas y ramas sería preferible hablar de nodos. Cada nodo es un hijo del nodo principal, excepto el del elemento raíz. Todos los nodos excepto las hojas también son nodos primarios.

Normalmente, dentro de la estructura de árbol, se utiliza el término técnico "padre" en lugar de "padres", ya que cada nodo solo tiene un padre.

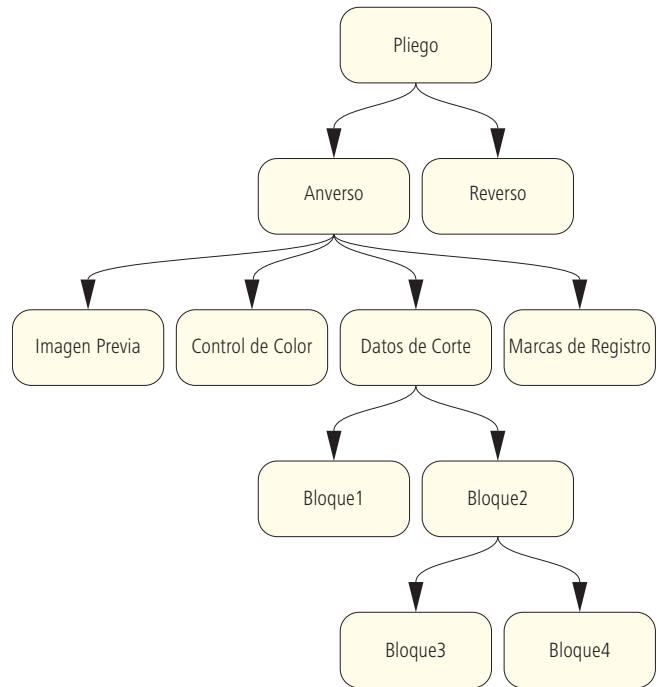
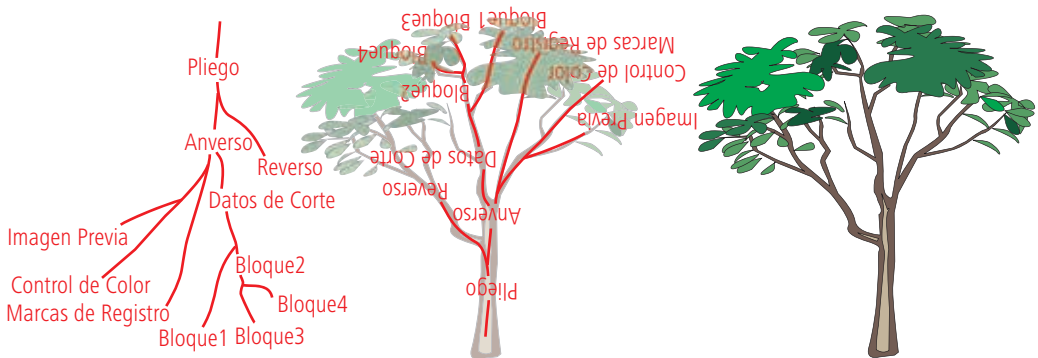


Figura 4.7 Estructura de árbol

La herencia es una importante característica en la estructura PPF y en otras estructuras de árbol.



Aunque los hijos heredan todas las propiedades del padre, también pueden tener sus propios valores y propiedades. Por ejemplo, los nodos "Anverso" (*Front*) y "Reverso" (*Back*) son hijos de "Pliego" y heredan sus propiedades, estas son el gramaje del papel (*CIP3AdmPaperGrammage*) y el tamaño (*CIP3AdmPaperExtent*). Quizás también hereden la propiedad "Tipo de pantalla" (*CIP3ADMTTypeOfScreen*), pero puede ser anulada por cualquiera de los nodos "Anverso" (*Front*) y "Reverso" (*Back*). Por último, el tipo de pantalla puede estar definida en los hijos y no aparecer en "Pliego".

Naturalmente, se pueden describir más "hojas" en PPF; pero serán hijos de otro elemento raíz llamado **Directorio PPF**.

La estructura de árbol en un archivo PostScript (PS) se realiza con sangrías. La figura 4.8 muestra la estructura de árbol de la figura 4.6 en este modo.

En la figura 4.9, vemos otro fragmento de código PPF. Las líneas primera y última, indicadas por símbolos de porcentaje, son comentarios y PostScript no las interpreta. Las líneas siguientes se definen a sí mismas. El formato descrito en puntos DPT tiene 1984,251968x2,54/72=70 cm de ancho y 1417,32283x2,54/72=50 cm de alto. El color del papel se define en CIEL*a*b y el peso en gr x m².

Figura 4.8
Ejemplo PPF

```

CIP3BeginSheet
  CIP3BeginFront
    CIP3BeginPreviewImage ...
    CIP3EndPreviewImage
    CIP3BeginColorControl ...
    CIP3EndColorControl
    CIP3BeginCutData ...
      CIP3BeginCutBlock ...
      CIP3EndCutBlock
      CIP3BeginCutBlock ...
        CIP3BeginCutBlock ...
        CIP3EndCutBlock
      CIP3BeginCutBlock ...
        CIP3BeginCutBlock ...
        CIP3EndCutBlock
      CIP3EndCutBlock
    CIP3EndCutData
    CIP3BeginRegisterMarks ...
    CIP3EndRegisterMarks
  CIP3EndFront
  CIP3BeginBack
  CIP3EndBack
CIP3EndSheet

```

El preajuste de zonas de entintado para prensas offset es la utilidad más conocida de PPF. El archivo PPF tiene un rol relativamente modesto porque solo guarda la previsualización (*Preview*) de los pliegos ensamblados y separados entre otras informaciones adicionales que son las que permiten a los programas especializados el cálculo de las zonas (ver figura 1.3).

Los valores de preajuste no forman parte de los estándares de PPF.

La figura 4.10 muestra el código para generar la imagen previa de la estructura PPF. Las imágenes están comprimidas y codificadas en ASC 1185.

```

%!PS-Adobe-3.0
...
CIP3BeginSheet
  /CIP3AdmArtist (Carl Cool) def
  /CIP3AdmJobCode (8) def
  /CIP3AdmJobName (8 Zustaende) def
  /CIP3AdmPaperExtent [ 1984.251968 1417.32283 ] def
  /CIP3AdmSheetName (FB 002) def
  /CIP3AdmTypeOfScreen (amplitude modulated) def
  /CIP3AdmPaperGrammage 100.0 def
  /CIP3AdmPaperThickness 0.035 mm def
  /CIP3AdmPaperColor [ 93.0 0.0 -3.0 ] def
  /CIP3AdmCreationTime (Mon Dec 11 18:29:57 2006) def
  ...
CIP3EndSheet
%%CIP3EndOfFile

```

El ancho (*CIP3PreviewImageWidth*) y alto (*CIP3PreviewImageHeight*) son en píxeles. La resolución está reducida a 50,8 píxeles por pulgada (*CIP3PreviewImagesResolution*).

Figura 4.9
Detalles de un archivo PPF

El nodo *CIP3PreviewImageMatrix* define la dirección de píxel de izquierda a derecha y de arriba abajo. La definición del cálculo matricial se puede encontrar en las especificaciones de PostScript y PDF (ver los ejercicios al final del capítulo 9).

Tras este paso siguen produciéndose cambios de tono en el pliego de manera que los datos de vista previa no son suficientes para el cálculo del preajuste de zonas de entintado. Entonces ¿qué o quién provoca estos cambios?

Figura 4.10
Definición de imagen
previa en un archivo PPF

```

CIP3BeginPreviewImage
  %%Page: 1
  %%PlateColor: Cyan
  CIP3BeginSeparation
    /CIP3PreviewImageWidth 1490 def
    /CIP3PreviewImageHeight 1210 def
    /CIP3PreviewImageBitsPerComp 8 def
    /CIP3PreviewImageComponents 1 def
    /CIP3PreviewImageMatrix [1490 0 0 -1210 0 1210] def
    /CIP3PreviewImageResolution [ 50.800 50.800 ] def
    /CIP3PreviewImageEncoding /Binary def
    /CIP3PreviewImageCompression /RunLengthDecode def
    /CIP3PreviewImageDataSize 515348 def
    CIP3PreviewImage...image data
  CIP3EndPreviewImage
  CIP3EndSeparation
  same for separations of Magenta, Yellow, and Black
CIP3EndPreviewImage

```

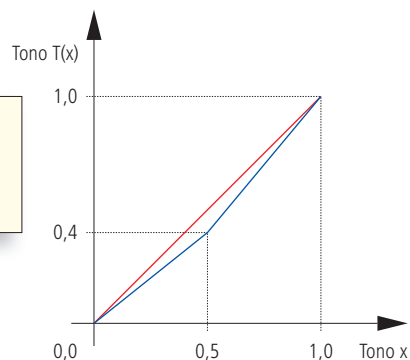
Para producir las planchas de impresión offset deben ser examinados los valores tonales y la profundidad de color de la imagen. Para obtener las tramas, generalmente, el tono solo se modifica una vez. Por ejemplo, la modificación suele ser igual al valor de ganancia de punto normalizada del soporte final. En argot PDF y PostScript se llaman **curvas de transferencia** o, según su finalidad, también pueden ser denominadas **curvas de linealización**, **curvas de calibración** y en flexografía **curvas de incremento** o **curvas de compensación de ganancia de punto**. La figura 4.11 muestra 2 curvas de transferencia y su código PPF. Los valores tonales siempre se muestran interpolados entre pares de coordenadas (denominadas x y $T_{(x)}$). La línea roja A es una curva de transferencia lineal sin efecto, la B reduce los valores de los tonos medios.

Figura 4.11
Curva de transferencia PPF

```

A  /CIP3TransferPlateCurveData
   [0.0 0.0 1.0 1.0] def
B  /CIP3TransferPlateCurveData
   [0.0 0.0 0.5 0.4 1.0 1.0] def

```



Un RIP puede proporcionar curvas de transferencia y previsualización, pero recibe las marcas de corte, registro y plegado o el color como datos gráficos irrelevantes para su función. El software de imposición sí puede interpretarlos y ensamblar la información en el PPF. En el archivo PPF se incluyen los datos de color y se posicionan las marcas de corte y registro en el pliego. Las marcas de corte también se pueden utilizar para definir los bloques de corte como vemos en la figura 4.12, donde se proporciona el tamaño de los paneles que guían el corte pero no se define la secuencia del corte. Por el contrario los datos de plegado sí permiten establecer la secuencia de plegado.

Hasta ahora todas las propiedades de PPF están relacionadas con el pliego de impresión. La versión 2.1 limitaba el área de aplicación del formato. Al publicarse la última versión de orden PPF 3.0 en 1998 permitió definiciones más amplias de productos. Incluso aceptaba la descripción de subproductos dentro de un producto global y se completó con la implantación de *Operaciones de Producto CIP3* (*CIP3Product-Operation*) y los *Parámetros de Producto CIP3*

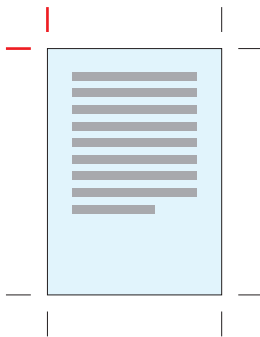
(/CIP3ProductParams), modelos de proceso/recurso presentados en el capítulo 3. En la versión 3.0 también se integraron operaciones como la recopilación o alzado, el grapado, el tipo de “cosido smyth” o el recorte con guillotina de tres cuchillas.

En los ejemplos PPF han aparecido 2 productores/emisores, el RIP y el programa de imposición y 3 consumidores/receptores, los módulos PPF de prensa, plegado y guillotina. Esto nos conduce a la teoría de imagen estructurada como ilustra la figura 4.13. Pero los distintos productores PPF lamentablemente, no generan un archivo PPF común. Ningún fabricante oferta una solución y la especificación PPF no establece directrices normativas para ello.

En la práctica, podemos encontrarnos con datos PPF diferentes dentro de la misma orden de trabajo y debemos aceptar la existencia de múltiples productores PPF, lo cual genera un flujo de trabajo mixto. Pero el consumidor necesita recopilar información PPF de distintos productores y la situación puede resultar caótica.

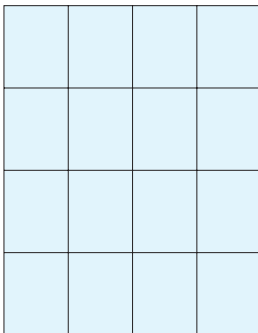
Figura 4.12
Alternativas para transmitir información de corte en PPF

Líneas para posicionar las marcas de corte



```
29.4 cm 5.0 cm /TopVerticalCutMark CIP3PlaceCutMark
29.4 cm 5.0 cm /LeftHorizontalCutMark CIP3PlaceCutMark
```

Bloque de corte alternativo



```
/CIP3BeginCutBlock
 /CIP3BlockTrf [1 0 0 1 1 4 cm 4 cm] def
 /CIP3BlockSize [29.9 cm 68.7 cm] def
 /CIP3BlockElementSize [10.1 cm 6.02 cm] def
 /CIP3BlockSubdivision [4 4] def
 /CIP3BlockType /CutBlock def
 /CIP3BlockElementType /Unknown def
 /CIP3BlockName (Block1) def
/CIP3EndCutBlock
```

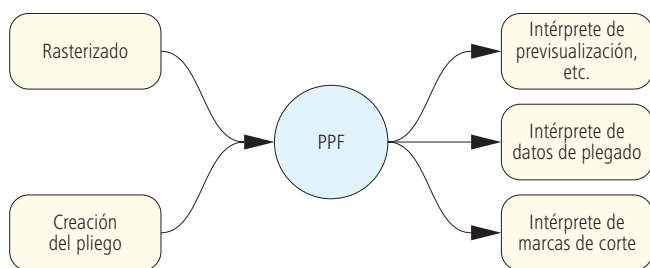


Figura 4.13
Productor/Consumidor PPF

Un ejemplo puede ser el interfaz de prensa que recibe la previsualización PPF desde el RIP y la posición de las marcas de registro desde el programa de imposición. Para el flujo de trabajo PPF es difícil mantener la funcionalidad dentro de esta complejidad creciente: gestionar tanto archivo PPF es complicado y se deben configurar varias carpetas activas por si se producen errores. En fin, no es extraño que el flujo PPF se limite a intercambiar los datos para calcular las zonas de entintado porque el coste no justifica la reducción del tiempo de preajuste en prensa y la consecuente reducción de residuos.

4.3 Formato de Orden de Trabajo Portátil (PJTF)

El **Formato Orden de Trabajo Portátil PJTF** [6] es el precursor del formato JDF. Al igual que PDF está estrechamente ligado a la arquitectura Extreme RIP, todos ellos desarrollados por Adobe Systems Incorporated.

La idea de la arquitectura Extreme es definir las funciones internas del RIP como módulos independientes alimentados por los metadatos necesarios que aporta PJTF. Los módulos se denominan **procesadores de órdenes de trabajo** (JTP) y son los encargados de convertir dichas órdenes de trabajo. Las tareas habituales de un RIP son Interpretación, Procesado y Tramado, pero los JTP no son solo esto, disponen de funciones como la normalización, las conversiones entre espacios de color, reventado, imposición, cálculo de datos para pruebas o la creación del archivo PPF para el preajuste de zonas de entintado. Los JTP pueden funcionar en un servidor o en un sistema compuesto por diferentes servidores.

El reventado JTP necesita mucho procesado y en muchos casos se realiza en un servidor externo por su gran consumo de CPU (procesador) y memoria RAM. A menudo se instalan varias instancias del JTP en uno o varios servidores para mejorar el rendimiento global. Durante años la arquitectura Extrema RIP ha sido la base de los sistemas CTP y de gestión de flujo de trabajo (WMS). Desde 2006 está siendo reemplazado gradualmente por la nueva tecnología de Adobe, **Print Engine**.

Los JTP se controlan desde un módulo *coordinador*. Este recibe las órdenes de trabajo desde el exterior y las distribuye a los distintos módulos de software *cliente* JTP.

Los datos transferidos más habituales son la definición del diseño de pliego y los ajustes del RIP. Un proveedor de sistemas Extreme RIP puede integrar sus propios JTPs o utilizar los de Adobe. En cualquier caso, el coordinador debe tener licencia de Adobe. El usuario no puede añadir JTPs externos al sistema. Por ejemplo, los JTP de otros fabricantes únicamente aportan los datos contenidos en el PJTF, no especifican ni el interfaz ni el funcionamiento del flujo de trabajo (ver figura 4.14).

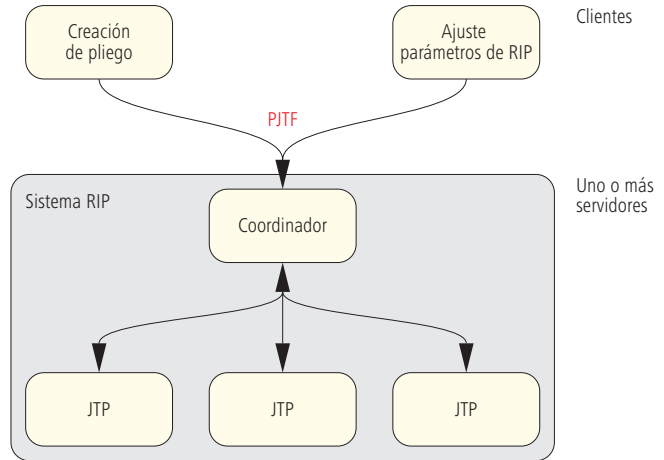


Figura 4.14
Arquitectura de Extreme RIP

A continuación exponemos los elementos que pueden declararse en PJTF:

- Datos de gestión de pedidos (fecha de impresión, dirección de entrega, etc).
- Valores de calibración para procesos del flujo de trabajo de salida (verificación previa, reventado, tramado, etc).
- Valores de calibración para procesos de acabado.

La atención del flujo de trabajo de salida CTP se centra principalmente en la imposición. Las demás áreas en la especificación PJTF están un tanto descuidadas.

Ya indicamos en secciones anteriores que XMP se codifica en XML y PPF en PostScript. PJTF se basa en PDF, por tanto es un archivo con estructura PDF y palabras clave específicas de PJTF. No obstante, Acrobat es incapaz de mostrarlas porque desconoce las palabras clave. Un archivo PDF es generalmente, una tabla de referencias cruzadas en la que se registra la situación exacta de una serie de objetos numerados. En la figura 4.15 vemos un objeto PJTF con el número 13, la descripción del objeto es muy similar al código PostScript/PPF. Contiene una tabla donde todas las líneas tienen la misma estructura: a la izquierda está la secuencia y a la derecha su valor, esta se puede interpretar como valor y palabra clave (nombre de variable) o como la entrada de un *diccionario* peculiar. De hecho Adobe los describe como "diccionarios". Cada *diccionario* está entre comillas angulares (<< y >>) igual que en PostScript y PDF.

```

13 0 obj
<<
  /BCL 0.95
  /ITO true
  /TW 0.2
  ...
>>
endobj

```

Figura 4.15
Objeto PJTF

El objeto 13 de la figura 4.15 describe los detalles del reventado. Las palabras clave están definidas en la especificación PJTF. Para el *Límite de Color Negro (BlackColorLimit)* utiliza */BCL* y el valor 0.95 indica que cada valor tonal superior a 95% es negro sólido. *Reventado Imagen a Objeto (ImageToObject-Trapping) /ITO* indica si las imágenes se deben expandir o contraer respecto a otros objetos. El *Ancho de Reventado (TrappingWidth)* lo define */TW* y es igual a 0.2 pulgadas.

PJTF también se estructura a modo de árbol como XMP y PPF. Los nodos se conectan con referencias entre objetos y estas se señalan con una "R" y un número delante del objeto al que hace referencia.

En este caso concreto, exploramos una rama que define un pliego. Cada rectángulo de la figura 4.16 representa un objeto del PJTF, cada flecha simboliza la referencia a otro objeto. Los números rojos encima del rectángulo son los identificadores de los objetos (ID).

El elemento raíz del PJTF es el objeto *Orden de Trabajo (JobTicket)* y hace referencia a 2 objetos: *Contenido de Orden (JobTicketContent)* y *Auditoría (Audit)*. Durante la producción los cambios realizados en el PJTF se registran en *Auditoría*. En *Contenido de Orden*, se encuentra el objeto *Signatura (Signature)* donde se combinan las hojas con la plantilla de imposición. En la figura no aparecen ni la posición de las marcas en el pliego ni la localización de los ficheros PDF en el sistema, referencias a *Contenido Orden de Trabajo*. El objeto 79 *Signatura* son los objetos 60 y 78 tipo *Hoja (Sheet)* que simbolizan las 2 hojas de impresión. El objeto hoja 60 hace referencia a 2 objetos tipo *Superficie (Surfaces)*, uno es el *anverso* 57 (lado fieltro o liso) y otro es el *reverso* 46 (lado tela o áspero). También está la referencia 59, *Origen de Material (MediaSource)* que define datos como tamaño, color de hoja, etc. Los marcadores de posición, tamaño y recorte se incluyen en el objeto 36 (*PlacedObject*).

En la figura 4.17 podemos ver el código de los objetos *Signatura* 79, 60, 46, y 38 detalladamente, las flechas rojas indican la situación de estas referencias. En la primera línea se identifica cada objeto y se abre el diccionario con <<. La segunda línea indica el tipo de objeto: *Signatura (Signature)*, *Hoja (Sheet)*, *Superficie (Surface)* u *Objeto Colocado (PlacedObject)*. Las demás entradas no serán detalladas porque son objetos específicos de sus diccionarios respectivos.

Comentaremos como ejemplo la entrada */CTM {0.0...0.0}* del Objeto 38 en donde se define el tamaño y la posición del objeto en la hoja.

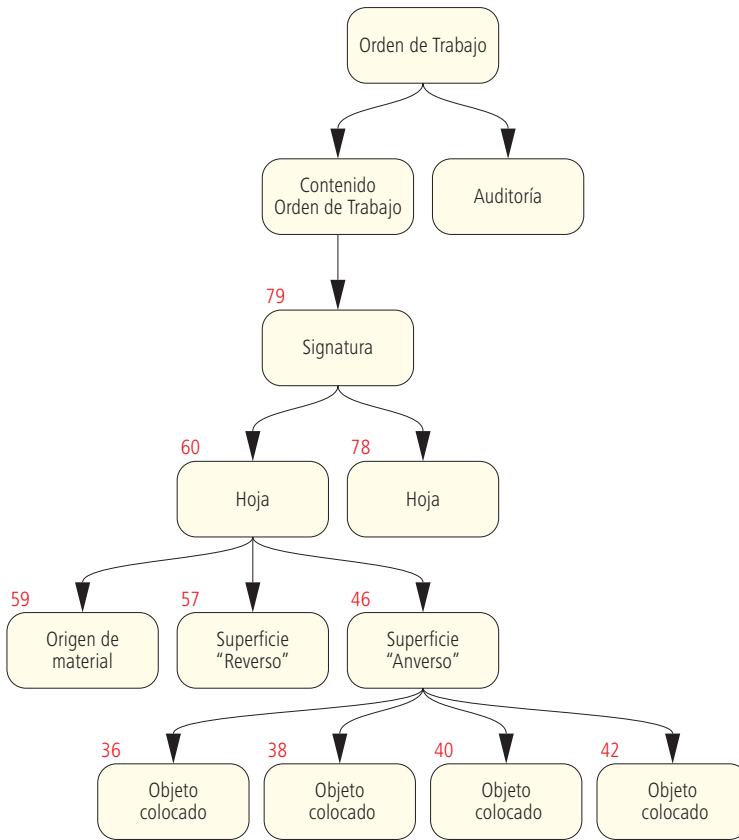


Figura 4.16 Estructura de un documento PJTF

Esta entrada se denomina **Matriz de Transformación Actual** o **Current Transformation Matrix CTM**.

Cada objeto y diccionario se cierra con la palabra *Fin de Objeto* (*endobj*) y el símbolo >>.

En resumen, podemos decir que es muy probable que JDF reemplazase totalmente a PJTF pero lo cierto es que en la actualidad muchos sistemas de flujo de trabajo todavía importan pliegos en PJTF.

```

79 0 obj <<
  /Type /Signatur
  /S{
    60 0 R
    78 0 R
  }
  >> endobj
...
60 0 obj <<
  /Type /Sheet
  /CP 1
  /Fr 46 0 R
  /B 57 0 R
  /MS 59 0 R
  ...
  >> endobj
...
46 0 obj <<
  /Type /Surface
  /PO{
    38 0 R
    42 0 R
    ...
  }
  >> endobj
...
38 0 obj <<
  /Type PlacedObject
  /D 0
  /O 0
  /CTM {0.0...0.0}
  /Cl{127.5...506.4}
  ...
  >> endobj

```

Figura 4.17
Referencias entre objetos

Ejercicios:

Abrir una fotografía digital en Photoshop y rellenar la información del archivo.

Colocar la imagen en InDesign.

Exportar el documento InDesign a PDF.

Examinar los datos XMP del archivo PDF en Acrobat. Seleccionar la imagen y ver las propiedades del menú contextual.

Examinar el archivo PDF utilizando un editor de texto como WordPad o una aplicación MFC (Microsoft).

Buscar "XMP" y examinar las entradas.

5 Breve introducción a XML

El **Lenguaje Extensible de Marcado XML (Extensible Markup Language)** no es un lenguaje, es más bien la descripción de un idioma; un metalenguaje o un sistema de definición de lenguaje. XML también es un estándar para intercambiar documentos entre programas. El Formato de Descripción de Trabajo JDF es un ejemplo de documento XML y por esta razón estamos hablando de XML. En este capítulo no lo trataremos en profundidad, solo presentaremos la terminología necesaria para comprender JDF. No lo comparamos con el **Lenguaje Estándar de Marcado Generalizado SGML (Standard Generalized Markup Language)** ni con el **Lenguaje de Marcado de Hipertexto HTML (Hypertext Markup Language)**. Tampoco hablaremos del beneficio o perjuicio de su uso en detrimento de otro tipo de documentos como PostScript o PDF. Encontraremos una presentación más precisa en [42].

En la sección 5.1 crearemos un documento XML y explicaremos los *elementos*, *atributos* y *valores*. La sección 5.2 trata la definición de las opciones (“vocabulario” XML) combinado con el *Espacio de Nombres XML (namespaces* o conjunto de nombres únicos). La sección 5.3 define el estándar basado en XML **cXML Lenguaje Extensible para Transacciones Comerciales (Commerce Extensible Markup Language)** componente básico para el intercambio electrónico de datos comerciales de la especificación JDF.

5.1 Construcción de documento XML

Habitualmente, la “declaración” es el prólogo de los demás elementos de un archivo XML. Se reconoce por estar contenida entre paréntesis angulares (< y >) o signos de interrogación (?). También aparece la versión, en este caso es la XML 1.0 (figura 5.1). Normalmente se utiliza la versión 1.0 aunque en el año 2004 se publicó la 1.1. Esta se compone del nombre o **atributo** *version* y su valor entre comillas “1.0”. Determinar el valor del atributo con un signo igual (=) se denomina **asignación de valores**.

Cuando se habla del atributo nos referimos solo al nombre, por ejemplo “versión”. Los distintos atributos se separan entre sí por un espacio.

Los demás atributos son opcionales pero la *versión* siempre debe estar en la declaración XML. El documento del ejemplo contiene otro atributo opcional, es el tipo de codificación denominado *encoding*.

```
<?xml version="1.0" encoding="UTF-8" ?>
```

Figura 5.1
Declaración XML

En este caso es UTF-8, **UTF Formato de Conversión UCS** y **UCS** es **Conjunto Universal de Caracteres Codificados en Octetos**. Está basado en la norma internacional *ISO/IEC-10646* que forma parte del conjunto de caracteres UNICODE y su longitud es de 8-bits como mínimo.

En la mayor parte de estructuras XML, tras la declaración está el *elemento raíz*. En JDF siempre se denomina *JDFroot* como veremos en el próximo capítulo. Los ladrillos de la estructura XML son los **elementos**. Ahora nos centraremos en el elemento raíz *Contacto*. Cada elemento tiene su nombre, como *Canal de Comunicación* (*ComChannel*), *Persona* (*Person*) o el mismo *Contacto* (*Contact*). Los elementos están delimitados por etiquetas de inicio y fin identificadas por paréntesis angulares. Por tanto, la etiqueta inicial del elemento será `<Contacto>` y la de cierre `</Contacto>`. La barra inclinada representa la etiqueta final (`/`). Los elementos, a su vez, pueden estar compuestos por subelementos, *Persona* es un subelemento o **elemento hijo (child element)**. Los 3 elementos *ComChannel* son *elementos hijo* de *Persona* y se suelen denominar **hermanos (siblings)**. En la figura 5.3 se puede apreciar claramente la estructura de árbol del código mostrado en la figura 5.2.

Los elementos sin subelementos se identifican como “vacíos” y no tienen etiquetas de inicio y fin, sin embargo, deben contener una barra inclinada (`/`) antes del paréntesis de cierre. El elemento *Empresa* (*Company*) no contiene subelementos adicionales y puede escribirse sin etiqueta final. Solo pueden escribirse así los elementos vacíos `<Elemento.../>`.

Los elementos vacíos también pueden tener sus propios atributos como vemos en el caso *Contacto* del ejemplo, aunque no todos los tienen. La especificación del tipo de documento define qué propiedad es un atributo y cuál es un subelemento.

Figura 5.2
Elementos XML

```
<Contact>
  <Person FirstName="Carl" FamilyName="Cool" NamePrefix="Herr">
    <ComChannel Locator="03475/101010" ChannelType="Phone"
      ChannelUsage="Private" ChannelTypeDetails="Landline" />
    <ComChannel Locator="03475/101011" ChannelType="Phone"
      ChannelUsage="Business" ChannelTypeDetails="Landline" />
    <ComChannel Locator="carl.cool@frisch.de"
      ChannelType="E-Mail" ChannelUsage="BusinessPrivate" />
  </Person>
  <Company OrganizationName="Frisch GmbH"/>
  <Address City="Eisleben" Street="Am Kaltenbach 3" Country="Deutschland"
    PostalCode="06295" />
  <!--Esto es una estructura simple JDF y código JDF a la vez, -->
</Contact>
```

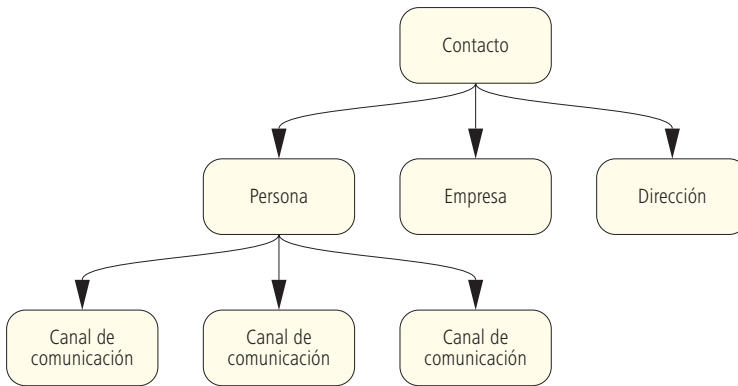


Figura 5.3
Elementos XML en
Estructura de árbol

Si el elemento *Contacto* es atributo del elemento *Persona* puede ser eliminado porque resulta inútil. Los elementos y atributos distinguen entre mayúsculas y minúsculas. Además, en los nombres se permiten todos los caracteres; ya sean específicos de XML, como el paréntesis angular o el apóstrofo u otros, como los espacios.

Un documento XML puede contener comentarios legibles para personas que el software XML no procesa, pueden ser varias líneas contenidas entre los símbolos de inicio "`<!--`" y fin "`-->`".

5.2 Espacio de Nombre XML

El formato XML permite intercambiar enormes cantidades de datos. Pueden ser desde la composición de una molécula, hasta una página descrita en **XML Paper Specification** de **MicrosoftXPS** o los documentos de un *Intercambio Comercial (Business2Business)* como facturas, pedidos y catálogos.

¿Cómo se definen en XML las estructuras de elementos y atributos? Lo cierto es que no hay reglas, cada uno puede definir su propio modelo con el **lenguaje de marcado**. Este se establece para intercambiar datos XML entre programas y se declara en el archivo con el **espacio de nombres XML**. Por ejemplo, si el documento XML es de uso interno no se producirán conflictos porque los elementos o atributos no se repiten. Pero en los documentos JDF es necesario declararlo para que la estructura respete la especificación CIP4 y designar los complementos propios del proveedor ajenos al flujo de trabajo.

El Espacio de Nombre XML se declara en el atributo *xmlns*. Su valor es un **Identificador Universal de Recursos URI (Universal Resource Identifier)**. Aunque parezca una dirección de internet, realmente se refiere al Espacio de Nombre (figura 5.4).

Tras el identificador URI puede haber un recurso de internet pero no es necesario. En otras palabras, puede ser un **Localizador Universal de Recursos** o dirección **URL (Uniform Resource Locator)**, pero generalmente no lo es. En el ejemplo, se puede ver lo fácil que es confundirlos: la primera es una dirección de internet URL y la segunda es un identificador URI y la dirección no existe.

El atributo *xmlns* de la segunda declaración está separado del prefijo HdM por 2 puntos (Hochschul der Medien es Universidad de Medios de Comunicación Stuttgart). Con la ayuda de estos prefijos podemos asociar nombres de elementos o atributos a los espacios de nombre. Cuando no contiene prefijo el Espacio de Nombre se considera por defecto.

En el ejemplo, el elemento *Fondo de Recurso (ResourcePool)* es subelemento de JDF y ambos nombres de elementos, al no contener prefijo, usan el espacio de nombre predeterminado *http://www.CIP4.org/JDFSchema_1_1*, mientras que el elemento privado *HdM:PrivateElement* está asociado mediante el prefijo *HdM* con el segundo espacio de nombres.

La estructura de un documento XML se puede determinar con una **Definición Tipo de Documento DTD (Document Type Definition)** o un Esquema (**XML Schema**); el esquema ofrece más posibilidades y es más moderno, pero para comprender este libro no es importante saber cuando utilizar uno u otro.

Los esquemas se conocen como **Definición de Esquema XML (XML Schema Definition)** y se utiliza su abreviatura XSD como terminación de archivo, por lo tanto "JDF.xsd" contiene el esquema JDF. Puede descargarlo de la página web de CIP4. Un esquema está codificado como texto y definido en XML. Esto significa que puede ser leído y editado con un programa de edición de textos y ser visualizado en cualquier navegador web.

Figura 5.4
Documentación del espacio
de nombre en XML

```
<?xml version="1.0" encoding="UTF-8" ?>
  <JDF xmlns="http://www.CIP4.org/JDFSchema_1_1"
    xmlns:HdM="www.hdm-stuttgart.de.com/schema/HdM"... >
    <ResourcePool... >
    ...
  </ResourcePool >
  <HdM:PrivateElement... />
</JDF>
```

A continuación enumeramos las definiciones que puede contener un esquema:

- Elementos autorizados para su utilización en el documento.
- Atributos permitidos en dichos elementos.
- Atributos requeridos u opcionales para un elemento.
- Relación de parentesco padre-hijo entre elementos.
- Tipo de datos de elementos y atributos
- Cardinalidad de elementos. (NdT. “Cardinalidad”: define el número de instancias referidas a la instancia de otra entidad).
- Referencias entre elementos.

Los documentos XML deben estar bien contruidos y validados. “Bien contruidos” significa que cumple las reglas sintácticas XML. Cuando cumple las normas adicionales del esquema está validado. El documento XML debe definir bajo que esquema se estructura.

Un **analizador** o **parser** es un programa que lee y analiza si el texto XML está bien formado. También se denomina **analizador de validación** o **validating parser**, porque verifica su validez. Esta función detecta si los documentos no están contruidos de acuerdo al esquema y los separa de los demás, si fuera necesario, reduciendo el riesgo de procesar documentos XML erróneos.

Para XML, un **analizador de validación** es como un programa de verificación previa para PDF (*PreFlight*).

5.3 Marco de Descripción de Recursos (RDF) y XMP

En el capítulo anterior ya tratamos y explicamos como escribir XMP con XML. El lenguaje **Marco de Descripción de Recursos RDF (Resource Description Framework)** [44] también se usa en internet para describir recursos, almacenar metadatos o realizar búsquedas exhaustivas para obtener información que no se encontraría de otro modo.

Además, al trabajar sobre el concepto de Semántica Web pueden leer y evaluar la metainformación de los motores de búsqueda y de los denominados Agentes Web.

Aunque la descripción de recursos RDF no es esencial para JDF, que-remos hacer una breve incursión en la presentación de datos XMP.

Si examinamos detenidamente la figura 5.5, veremos como se muestra un objeto RDF/XMP en formato PDF: el elemento raíz es `<x:xmpmeta>` y tiene un único elemento hijo `<rdf:RDF>`, este hace referencia a 3 elementos hijo `<rdf:Description>` a su vez. Cada elemento `rdf:Description` define un espacio de nombre independiente: el primero con el prefijo `pdf`, se guarda en archivos PDF; el segundo con prefijo `xap` está relacionado con los datos del archivo, pasando a autoría una vez modificado y el tercero `dc` "Dublin Core" hace referencia al título y autor del archivo (ver capítulo 4).

Todavía nos queda alguna línea por aclarar: en este caso el elemento `<rdf:Seq>` describe una lista ordenada que consta de una entrada; cada entrada es un elemento `<rdf:li>`. Se pueden especificar valores alternativos mediante el elemento `<rdf:Alt>`.

Figura 5.5
Código XMP en XML

```
20 0 obj
...
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP toolkit 2.9-9, framework
1.6">
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:x="http://ns.adobe.com/iX/1.0/">
    <rdf:Description rdf:about="" xmlns:pdf="http://ns.adobe.com/pdf/1.3/"
      pdf:Trapped="False" pdf:Producer="Acrobat Distiller 7.0 (Windows)"
      pdf:GTS_PDFXConformance="PDF/X-1a:2001"
      pdf:GTS_PDFXVersion="PDF/X-1:2001">
    </rdf:Description>
    <rdf:Description rdf:about="" xmlns:xap="http://ns.adobe.com/xap/1.0/"
      xap:CreateDate="2007-08-24T12:00:39+02:00"
      xap:CreatorTool="PScript5.dll Version 5.2"
      xap:ModifyDate="2007-08-24T12:00:39+02:00">
    </rdf:Description>
    <rdf:Description rdf:about=""
      xmlns:dc="http://purl.org/dc/elements/1.1/">
      <dc:creator>
        <rdf:Seq>
          <rdf:li>Carl Cool</rdf:li>
        </rdf:Seq>
      </dc:creator>
      <dc:title>
        <rdf:Alt>
          <rdf:li xml:lang="x-default">
            test.indd
          </rdf:li>
        </rdf:Alt>
      </dc:title>
    </rdf:Description>
  </rdf:RDF>
</x:xmpmeta>
...
endobj
```

También aparece un atributo que queremos destacar *rdf:about=""*. El atributo *RDF* se desarrolló inicialmente para describir recursos de Internet. El valor del atributo "*about*", proporciona la dirección URI del recurso correspondiente en internet. En el ejemplo XMP, encontramos una secuencia vacía donde los metadatos no corresponden a un recurso de internet; se refieren a los metadatos de un PDF localizado en el sistema de archivos local que guarda la información XMP.

5.4 Lenguaje de Marcado Extensible para Comercio (cXML)

En la vida empresarial se producen muchos intercambios de documentos entre clientes y proveedores. Se envían y reciben constantemente catálogos de servicios y productos, ofertas solicitadas, propuestas de pedido, órdenes de trabajo, confirmaciones y facturas. Estos intercambios también se realizan por carta, correo electrónico o mediante sistemas de comercio electrónico o portales de internet.

Para simplificar, podemos decir que cliente e imprenta son socios en la industria de la comunicación gráfica. Existen diferentes enfoques para describir estas operaciones comerciales.

CIP4, responsable del Formato de Definición de Trabajo JDF, distribuye para este fin un Lenguaje de Marcado XML denominado **PrintTalk®**. Actualmente está vigente la versión 1.3 pero ya se está trabajando, en paralelo a la especificación JDF 1.5, en el borrador de la versión 1.5 de PrintTalk.

JDF describe el producto y PrintTalk las transacciones comerciales. Hablando claro, JDF se puede integrar en la transacción PrintTalk. Cada uno tiene tareas específicas, aunque ambos lenguajes XML están estrechamente ligados. Esto permite la implementación de modelos de comercio electrónico como "Web to Print" donde los clientes pueden elegir y personalizar sus productos de impresión, comunicar los datos empresariales y realizar el pedido.

Trataremos PrintTalk detalladamente en la sección 8.5, pero está basado en **cXML Lenguaje de Marcado para Transacciones Comerciales (Commerce Extensible Markup Language)** y es necesario describir sus funciones brevemente.

Los documentos cXML contienen las transacciones comerciales y básicamente se crean como vemos en la Figura 5.6. El elemento raíz siempre es un elemento cXML. Este contiene los elementos *Cabecera* y *Solicitud*. El elemento de cabecera alberga la información para atender al remitente y al receptor.

```

<cXML>
  <Header>
    Información Cabecera
  </Header>

  <Request>
    Información Solicitud
  </Request>
</cXML>

```

Figura 5.6
Estructura cXML
Figura 5.7
Documento cXML

Los datos de identificación del remitente pueden contener una contraseña. El marco para la cabecera se muestra en la Figura 5.7. Los elementos *Desde* (*From*) y *A* (*To*), sirven para identificar al remitente y al receptor del documento cXML. El *Remitente* y *Desde* pueden ser el mismo pero no tiene por qué ser así. Pueden ser diferentes porque el elemento *Desde* define al remitente lógico del documento, pero si el documento cXML se canaliza mediante redes de comercio electrónico, el elemento *Remitente* está definido en la instancia establecida por el http de unión al receptor.

Si es necesario también se puede introducir una contraseña en el elemento *Remitente* (*Sender*). El elemento *Solicitud* es la petición del artículo. También se pueden introducir en *OrderRequestHeader* las funciones administrativas, como la dirección de entrega y facturación o la información sobre pago de impuestos. El elemento *Artículos Solicitados* (*ItemOut*) contiene los detalles sobre la cantidad de artículos pedidos.

La *Cabecera* (*Header*) de *PrintTalk* es idéntica a la de cXML y el elemento *Solicitud* también se puede encontrar en *PrintTalk*.

También vemos que el elemento hijo *OrderRequest* no aparece, en su lugar encontramos un *BusinessObject* que describe la transacción, por ejemplo una oferta, solitud de presupuesto o similar.

También podemos encontrar otros elementos hijo de *BusinessObject* que almacenan la información sobre el producto de impresión solicitado.

```

<cXML>
  <Header>
    <From>
      ...
    </From>

    <To>
      ...
    </To>
    <Sender>
      ...
    </Sender>
  </Header>

  <Request>
    <OrderRequest>
      <OrderRequestHeader ...>
        ...
        <ShipTo>
          ...
        </ShipTo>
        <BillTo>
          ...
        </BillTo>
        <Tax>
          ...
        </Tax>
        <Payment>
          ...
        </Payment>
      </OrderRequestHeader>
      <ItemOut ...>
        <ItemID>
          ...
        </ItemID>
      </ItemOut>
    </OrderRequest>
  </Request>
</cXML>

```

6 Introducción a JDF

Ahora, resumimos los conceptos más relevantes para JDF aprendidos en capítulos anteriores:

- JDF es un formato de Orden de Trabajo que puede almacenar datos técnicos y funciones de gestión de pedidos.
- JDF incluye funciones PPF y PJTF.
- JDF está basado en el modelo proceso/recurso.
- JDF está basado en XML.

En la primera sección del capítulo 6, presentaremos el código JDF detalladamente. Estos apartados representan la base para todos los capítulos posteriores. En la última sección haremos hincapié en los flujos de trabajo, en particular en las **Especificaciones de Conformidad e Interoperabilidad ICS (Interoperability Conformance Specifications)**.

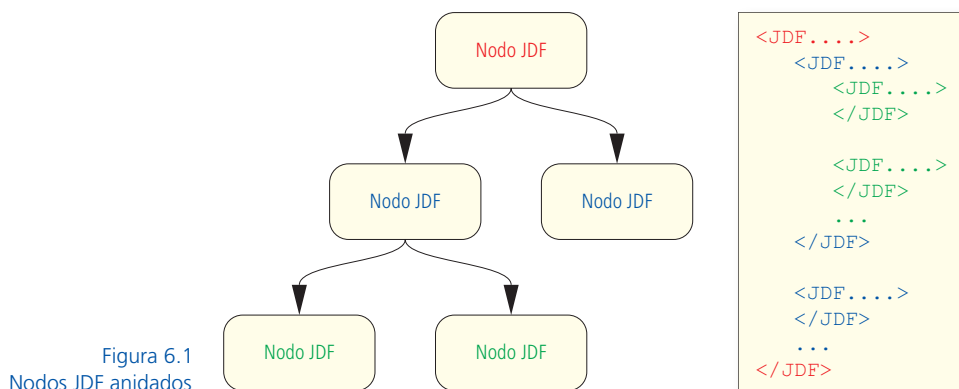
Esta introducción a JDF debe incluir un resumen de los procesos de producción posibles, con descripciones de producto y especificaciones. Al final del capítulo, debería ser capaz de leer y comprender un documento típico JDF aunque debemos tener claro que este capítulo no puede sustituir a la especificación JDF mucho más detallada.

6.1 Construcción de un documento JDF

En el flujo de trabajo JDF por cada trabajo de impresión debe haber al menos un documento JDF con los metadatos necesarios para realizar el producto. Como es previsible en un documento XML el documento JDF tiene estructura de árbol. Se denominan **nodos JDF** cada elemento raíz del árbol, cada rama y cada hoja. Normalmente un trabajo de impresión se compone de varios nodos JDF donde cada nodo describe una porción del trabajo y cada uno de ellos es un elemento XML que puede tener subelementos adicionales.

Naturalmente, no es necesario que tengan los tres niveles jerárquicos de la Figura 6.1, puede haber más o menos niveles. Un nivel puede contener un número indeterminado de nodos.

Tan solo unos pocos nodos JDF representan un proceso. Generalmente son las hojas del árbol, respecto a esto la estructura de datos JDF se basa parcialmente en el modelo de proceso/recurso.



Estos 4 nodos JDF diferentes se denominan:

- Nodo producto (product intent node).
- Nodo proceso de grupo.
- Proceso combinado.
- Nodo de proceso.

Un producto manufacturado y todas sus partes constituyentes o subproductos se definen utilizando el **nodo producto**. Por ejemplo, un libro se realiza con la cubierta y el contenido; el resultado son 3 nodos producto JDF: el Libro, la Cubierta y el Contenido.

Los **nodos de proceso** definen las tareas necesarias para obtener el producto impreso o alguno de sus componentes. Ya conocemos el **proceso combinado** de interpretar, rasterizar, tramar, cortar, plegar, recopilar, grapar, guillotinar y apilar. Obviamente existen muchos más procesos, pero ya los iremos tratando en los siguientes capítulos.

Los procesos pueden ser reunidos o combinados por diferentes razones. Este tipo de nodos JDF se denominan **nodos proceso de grupo** y pueden utilizarse para combinar procesos según las necesidades.

Pongamos como ejemplo la figura 3.8 Rasterizado en RIP. Puede ser tanto un proceso de grupo como un proceso combinado compuesto por los procesos de interpretación, rasterizado y tramado (ver figura 3.10).

En la sección 6.4 nos referimos a las diferencias concretas entre los nodos de proceso y de proceso combinado.

La figura 3.4 Grabar Planchas podría ser un ejemplo válido de nodo proceso de grupo. Contiene la imposición, la filmación y un nuevo proceso, rasterizado. También tenemos un grupo adicional que incluye a su vez otros procesos y procesos de grupo. Se podría describir como los objetos gráficos previamente agrupados en los programas de dibujo vectorial que se vuelven a agrupar.

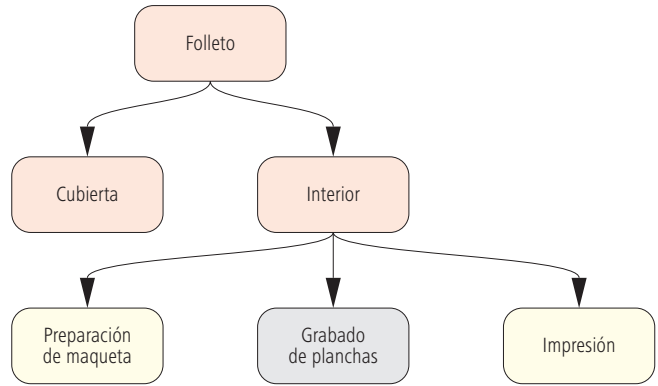


Figura 6.2
Relación entre tipos de nodo JDF:
Nodo Producto rojo
Nodo Proceso de grupo gris
Nodo Proceso amarillo

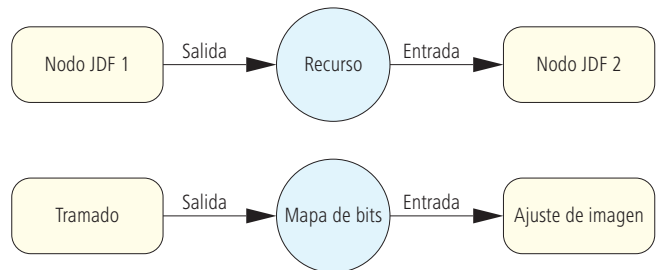
La figura 6.2 no pretende ser una explicación en profundidad, solo es un ejemplo de relación entre tipos de nodo JDF. Los nodos coloreados en rojo simbolizan nodos de producto, los grises son nodos proceso de grupo y los amarillos nodos de proceso. No incluye ningún proceso combinado.

Debemos señalar que estos 3 tipos de nodo no tienen por qué aparecer en cada documento JDF. Por lo tanto, es muy posible que un documento JDF pueda estar formado por un solo nodo, ya sea de producto o de proceso de grupo. Sin embargo, un documento JDF que contenga un pedido completo de impresión no está constituido exclusivamente por nodos de proceso, ya que los procesos siempre se refieren a partes de productos o al producto en sí. En el caso del ejemplo, el elemento raíz es un nodo producto JDF que representa el trabajo completo.

Durante la preparación de un producto impreso, el documento JDF va creciendo a medida que circula a través del flujo de trabajo. Si aún no están definidos todos los nodos al principio del proceso, los datos se irán incorporando gradualmente. Las inserciones se producen con las entradas que realiza el operario durante la producción o mediante valores predefinidos de producción, como se describe en la sección 3.2.

Figura 6.3
Relación entre recursos y los tipos de nodo JDF

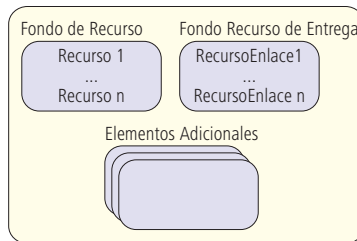
Los nodos JDF no son entes aislados y necesitan incorporar recursos. Los recursos son las entradas o salidas de los nodos (Sección 3.2). Un **recurso** puede ser electrónico (archivos, parámetros, etc) o físico (papel, planchas, etc).



Este principio se describe de forma sencilla en la figura 6.3. Para describir un flujo de trabajo existe una compleja red de nodos y recursos, como ya vimos en el capítulo 3 (figura 3.8).

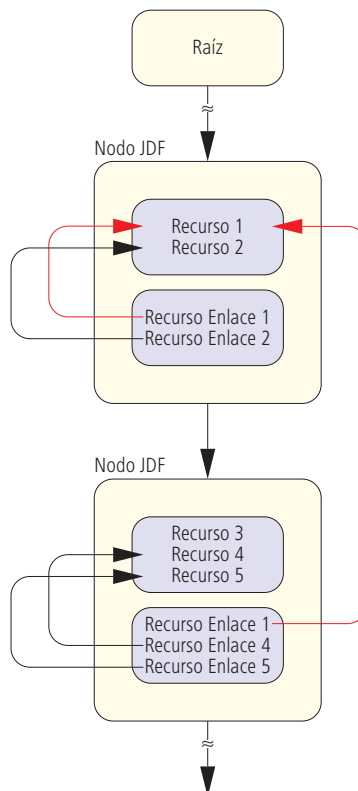
Los nodos de proceso, ciertamente, no son los únicos con recursos. Los nodos de producto también los tienen, por ejemplo, describen detalles como la asignación de papel para un subproducto.

Figura 6.4
Nodo JDF y subelementos



Por tanto, debemos adoptar la mentalidad intuitiva patente en los procesos, el modelo productor/consumidor. Es fácil entender que el proceso impresión “consume” planchas, pero nos resulta difícil de asimilar que un nodo de producto “consume” nodos descriptivos de papel, como un sobre o una tarjeta.

Figura 6.5
Dos nodos JDF accediendo al mismo recurso



¿Cómo se estructura la relación entre recursos y nodos JDF? Cada nodo JDF no solo alberga atributos y valores, también contiene subelementos.

Existen dos subelementos, en particular, especialmente importantes: el *Fondo de Recurso* (*ResourcePool*) y el *Fondo de Enlace de Recursos* (*ResourceLinkPool*).

El primero de ellos alberga los *Recursos* para los nodos JDF y el segundo los *Enlaces del Recurso* para los nodos JDF como son los nodos de entrada o salida según sea lo requerido (figura 6.4).

No es preciso que estén definidos en *ResourcePool* todos los recursos de un nodo JDF. Aunque estén contenidos en el *ResourcePool* de otro nodo JDF también podemos usar los que aparecen en lo alto del árbol, o sea, cerca de la raíz.

Por lo tanto, los procesos pueden compartir un recurso sin que esto signifique que se aplicará dos veces (Figura 6.5).

Ni siquiera es necesario que un nodo JDF tenga *ResourcePool*. Sin embargo, cada nodo JDF debe tener un *ResourceLinkPool* asociado, porque los *Enlaces de Recurso (ResourceLink)* siempre se refieren a los nodos JDF de los *ResourceLinkPool*.

Esto significa que si un nodo JDF no tiene *ResourceLinkPool* tampoco tiene relación con los recursos correspondientes. En la figura 6.6 vemos el elemento raíz y un subnodo, una estructura típica de documento JDF.

```

<!xml.....!>
<jdf.....>
  ...
  <ResourcePool>
  ...
  </ResourcePool>
  <ResourceLinkPool>
  ...
  </ResourceLinkPool>

  <jdf.....>
    ...
    <ResourcePool>
    ...
    </ResourcePool>
    <ResourceLinkPool>
    ...
    </ResourceLinkPool>
  </jdf>
  ...
</jdf>
    
```

Diagrama de la estructura de JDF con su elemento raíz y los subelementos. El código muestra un elemento raíz <jdf.....> que contiene un subnodo <jdf.....>. Los subnodos contienen <ResourcePool> y <ResourceLinkPool>. Las etiquetas <ResourcePool> y </ResourcePool> están agrupadas por un corchete y etiquetadas como 'Recursos'. Las etiquetas <ResourceLinkPool> y </ResourceLinkPool> están agrupadas por un corchete y etiquetadas como 'Relación con los recursos'.

Figura 6.6 Estructura de JDF con su elemento raíz y los subelementos

```

<?xml version="1.0" encoding="UTF-8" ?>
<JDF ID="_4711" DescriptiveName="Frisch Advertising" JobID="_42"
  Status="InProgress" Type="Product" Version="1.3"
  xmlns="http://www.CIP4.org/JDFSchema_1_1">
<!-- Generated by the CIP4 C++ open source JDF Library version
  CIP4 JDF Writer Java 1.3 -->
...
</JDF>
    
```

Figura 6.7 Atributo típico de un elemento raíz JDF

6.2 Ejemplos de Nodos JDF

En esta sección, examinaremos detenidamente los fragmentos de documento JDF que consideramos más ilustrativos. Hemos omitido muchos elementos y atributos para no sobrecargar la sección.

La figura 6.7 muestra la primera línea del prólogo XML tal y como establecimos en el capítulo anterior. El elemento raíz se denomina JDF y está marcado en rojo. En el código del ejemplo las denominaciones de atributos siempre se colorean en verde. Explicaremos brevemente los múltiples atributos de los nodos JDF:

- En el documento cada nodo se identifica con un número o ID.
- El valor del atributo opcional *Nombre* (*DescriptiveName*) es un identificador de nodo JDF legible para el ser humano. El nodo del ejemplo se denomina "Frisch Advertising" porque el documento JDF es un producto para la Agencia Frisch.
- El valor *Identificador de Trabajo* (*JobID*) es el ID asignado por el MIS.
- El estado del nodo JDF se define en el atributo *Estado* (*Status*). Los estados que podemos encontrar son: *En Curso* (*InProgress*), *Preparado* (*Ready*), *Detenido* (*Stopped*), *Terminado* (*Completed*) e *Interrumpido* (*Aborted*) entre otros.
- El nodo JDF tipo *Producto* (*Product*) no es un nodo de proceso ni de proceso de grupo (*ProcessGroup*) y tampoco de proceso combinado (*Combined*). El nodo de proceso identifica el tipo de proceso, como *Interpretar*, *Corte*, *Plegado* o similar.
- *Version* identifica la especificación JDF utilizada para la construcción del nodo JDF.
- El atributo *xmlns* es el Espacio de Nombre XML (sección 5.2).

Figura 6.8
Fondo de recurso con
información del cliente

A continuación de los valores y atributos aparece un comentario XML sobre el lenguaje de programación utilizado (C++) y la biblioteca concreta que se utilizó para crear el documento. (CIP4 JDF Writer Java 1.3).

```
<ResourcePool>
  <CustomerInfo ID="_4712"= CustomerID="_0815" Class= "Parameter"
    CustomerJobName="Frisch Advertising" Status="Available" />
    <Contact ContactTypes="Customer">
      <Person FirstName="Carl" FamilyName="Cool" NamePrefix="Herr">
        <ComChannel Locator="03475/101010" ChannelType="Phone"
          ChannelUsage="Private" ChannelTypeDetails="Landline" />
        ...
      </Person>
      <Company OrganizationName="Frisch GmbH" />
      <Address City="Eisleben" Street="Am Kaltenbach. 3"
        Country="Deutschland" PostalCode="06295" />
    </Contact>
    ...
  </CustomerInfo>
  ...
</ResourcePool>
```

Siguiendo con el ejemplo del folleto Frisch, tenemos el *Fondo de Recurso* y el *Fondo Recurso de Entrega* del elemento raíz. La figura 6.8 solo muestra el primer elemento de *ResourcePool*, donde se almacenan los datos de cliente *CustomerInfo*. Cada recurso, igual que cualquier nodo JDF, requiere un ID único. Además vemos que el atributo *Clase* (*Class*) contiene un recurso *Parameter en argot* JDF es decir los datos de la persona están en forma de identificación ID JDF y por último, el *Estado* está *Disponible* (*Available*). Si las direcciones de cobro y envío son diferentes se pueden añadir otras.

```
<Component ID="_4713" Class="Quantity" ComponentType="FinalProduct"
  DescriptiveName="Frisch Advertising" Status="Unavailable" />
...
<DeliveryIntent Class="Intent" ID="_4714" Status="Available">
  <DropIntent>
    <DropItemIntent Amount="2000">
      <ComponentRef rRef="_4713" />
    </DropItemIntent>
  </DropIntent>
</DeliveryIntent>
```

En *ResourcePool* vemos contenidos algunos recursos adicionales que describimos en la figura 6.9. Son los recursos *Componente* y *Preferencias de Envío*, *Component* y *DeliveryIntent* respectivamente. El recurso *Component* representa el producto final y es un requisito en cada elemento raíz JDF. *Quantity* (*Cantidad*) está definido como *Class* y es un posible subtipo de recurso físico, los productos finales se pueden contar, los recursos físicos como la tinta no. Cómo, cuándo y dónde se entrega el producto al cliente se describe en el recurso *Preferencias de Envío* (*DeliveryIntent*). En este caso se trata de una sola entrega (*DropIntent*) y la cantidad son 2000 copias, como describe *Preferencias Entrega de Artículo* (*DropItemIntent*).

La línea más interesante es, sin duda, la referente a *Component*; significa que el envío nos remite al producto final.

El atributo identificador del producto final es *rRef* = "4713" y generalmente hace referencia a un recurso.

Figura 6.9
Recursos del elemento raíz
Component y *DeliveryIntent*

```

<ResourceLinkPool>
  <DeliveryIntentLink Usage="Input" rRef="_4714" />
  <ComponentLink Amount="2000.0" Usage="Output" rRef="_4713" />
  <ComponentLink Usage="Input" rRef="_4715" />
  <ComponentLink Usage="Input" rRef="_4716" />
</ResourceLinkPool>

```

Figura 6.10
ResourceLinkPool

En la figura 6.10 podemos ver el *Fondo Enlace de Recursos* (*ResourceLinkPool*). En él encontramos las referencias entre nodos JDF y recursos, también nos indica cuales son recursos de entrada (*Input*) o salida (*Output*).

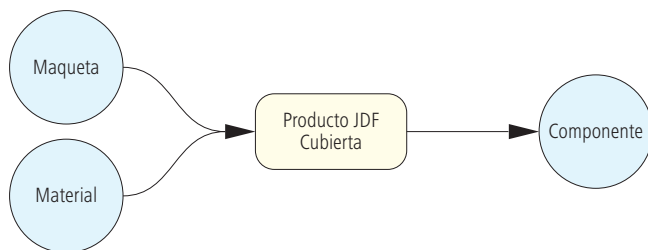
La relación entre el *Enlace de Recurso* (*ResourceLink*) y el recurso es sencilla. Si se define un *Enlace de Recurso* con el nombre XYZ y tiene el ID 333, el *Enlace de Recurso* se denominará XYZLink y se le asignará la referencia *rRef*="333". El ejemplo se ha preparado para que los recursos *DeliveryIntentLink* de entrada y salida sean evidentes

Todavía no hemos explicado los recursos de entrada 4715 y 4716 del ejemplo. El elemento raíz JDF tienen 2 subnodos JDF, el primero es la cubierta y el otro las páginas del producto. Ambos se describen detalladamente mediante recursos *Component*, porque para realizar el producto final son necesarias las páginas y la cubierta. Los componentes se convierten de este modo, en los recursos de entrada del producto final.

Cada uno de los subnodos JDF tienen su propios *ResourcePool* y *ResourceLinkPool*. También hay información sobre los otros subproductos: tipo y color del papel o número de páginas del subproducto.

El recurso *Preferencias de Maqueta* (*LayoutIntent*) nos da los detalles del tamaño final del producto y el número de páginas, las características del papel están descritas en el recurso *Preferencias de Soporte* (*MediaIntent*). Ver figura 6.11.

Figura 6.11
Subnodo JDF "Cubierta"
separado con su propio
Enlace de Recurso



Si atención al cliente o el director del proyecto son los encargados de insertar la información en el MIS cabe la posibilidad de que ciertos detalles queden incompletos. Por ejemplo, puede ser que desconozcan el número de páginas exacto o seleccionen un gramaje del papel aproximado. Por tanto, estos recursos necesitan un margen para cambios posteriores.

Los recursos *Intent* solo se encuentran en *nodos de producto*, también denominados *nodos de preferencias de productos*.

Por seguir con el ejemplo anterior, queda claro que los recursos definen por separado la cubierta y el contenido, no es una descripción del producto final. Debe hacerse así porque puede utilizarse diferente tipo de papel en cada uno de ellos o usan colores distintos.

```
<JDF DescriptiveName="Cover" ID="_4715" Type="Product" >
  <ResourcePool>
    <MediaIntent ID="_4717" Class="Intent" Status="Available"
      DescriptiveName="Paper for Cover">
      <Dimensions Actual="2437.7952755905512 1729.1338582677165"
        DataType="XYPairSpan"
        Preferred="2437.7952755905512 1729.1338582677165" />
      <Weight DataType="NumberSpan" Range="115 ~ 125" Preferred="120" />
    </MediaIntent>
  </ResourcePool>
</JDF>
```

La figura 6.12 muestra la cubierta con el recurso *MediaIntent* (Clase *Intent*). Debemos tener en cuenta que sus dos subelementos son *Medidas (Dimensions)* y *Pesos (Weight)*.

Figura 6.12
Recurso para la elección de material con información sobre el sustrato elegido.

El valor de los atributos *Actual* y *Preferido (Preferred)* es idéntico en los elementos *Dimensions*. Ambos son tamaños de papel en puntos DTP. Por ejemplo, para que las medidas sean comprensibles en centímetros se divide entre 72 y se multiplica por 2,54; el resultado es 86x61cm. *Preferred* es el valor seleccionado por el cliente y el elegido por la imprenta *Actual*, obviamente son iguales.

El peso del papel elegido por el cliente se registra en el atributo *Weight* y es en gr/m^2 . Vemos que el *Rango (Range)* está entre 115 y 125g/m^2 y el valor seleccionado en *Preferred* es de 120g/m^2 .

El analizador XML (*Parser*) debe ser informado de que el rango de números estará entre "115~125". Esto sucede porque el tipo de datos se define como *Intervalo de Número (NumberSpan)*, la *Dimensión (Dimensions)* usa el mismo tipo de datos *XYPairSpan* que significa *Serie de NúmerosXY*: El tamaño de papel es Ancho1xAlto1 a Ancho2xAlto2 sin embargo en la figura 6.12 no se utiliza sino que se da un tamaño de papel fijo.

Además de *ResourcePool* y *ResourceLinkPool*, cada nodo JDF puede contener otro grupo ignorado hasta ahora *AuditPool*. Este registra los pasos empleados para realizar el impreso, desde las modificaciones del documento JDF hasta la información del proceso JDF.

Todas las operaciones quedan almacenadas en el *Fondo de Auditoría (AuditPool)*, como el cambio de planchas, lavado de mantillas, impresiones ok, etc. Esto ocurre obviamente, si los pasos se van registrando correctamente en la planta de impresión.

A continuación, mostramos una lista general pero incompleta de los *Elementos de Auditoría (Audit Elements)* disponibles en los *Fondos de Auditoría (AuditPool)*:

- Creación, cambio o borrado de nodos.
- Tiempos de proceso (Inicio, final...).
- Estado final (Completado, Cancelado, Detenido...).
- Error.
- Recursos utilizados o ausentes.

El MIS puede analizar los costes del trabajo impreso mediante los contenidos de *AuditPool*.

La figura 6.13 muestra un *AuditPool* básico donde las únicas entradas son quién creó el nodo y dónde:

- *AgentName* es el nombre del programa que creó el nodo JDF.
- *AgentVersion* es la versión de este programa.
- *Author* es la persona o usuario del PC que creó el nodo.
- *TimeStamp* es el momento de generación de los nodos.

Figura 6.13
Fondo de Auditoría
(AuditPool)

El nodo se creó el 22 de Octubre del 2008 a las 17.09 horas. El valor 47 son los segundos y +01:00 es la diferencia horaria respecto al meridiano de Greenwich.

```
<AuditPool>
  <Created AgentName="SuperJDF" AgentVersion="1.0" Author=" Administrator"
    ID="_4717" TimeStamp="2008-10-22T17:09:47+01:00" />
</AuditPool>
```

6.3 División de Recursos

Ciertos recursos plantean un problema. Por ejemplo, si vamos a imprimir en cuatricromía, ¿las 4 planchas son un recurso o cada plancha necesita el suyo?. La situación se complica al especificar el tipo de papel utilizado en las tapas y páginas interiores.

```
<Component ID="_4713" Class="Quantity" ComponentType="FinalProduct"
  DescriptiveName="Frisch Advertising" Status="Unavailable"
  PartIDKeys="Condition">
  <Component Condition="Good" IsWaste="false" />
  <Component Condition="Waste" IsWaste="true" />
</Component>
```

Tratar cada subproducto en conjunto tiene sentido, pero de vez en cuando es necesario contar las hojas sueltas, por ejemplo, para saber cuanto desperdicio hemos generado.

Figura 6.14
Partición de recursos

Asignar un recurso JDF por hoja sería absurdo y el tratamiento de los documentos resultaría imposible. Lo mejor es tener un recurso para todas las hojas de impresión, pero que se pueda dividir en componentes individuales (particiones). Por lo tanto, necesitamos recursos divisibles.

Ahora, vamos a explicar como se estructura este tipo de recursos. Ya fue mostrado en la figura 6.9 el recurso *Componente*, pero a partir de ahora lo trataremos como recurso divisible. En la figura 6.14 podemos ver que la única novedad del *Componente ID 4713* es el atributo *PartIDKeys*. El valor asociado contiene la regla de divisibilidad o, para exponerlo con más claridad, es la *clave de la partición*.

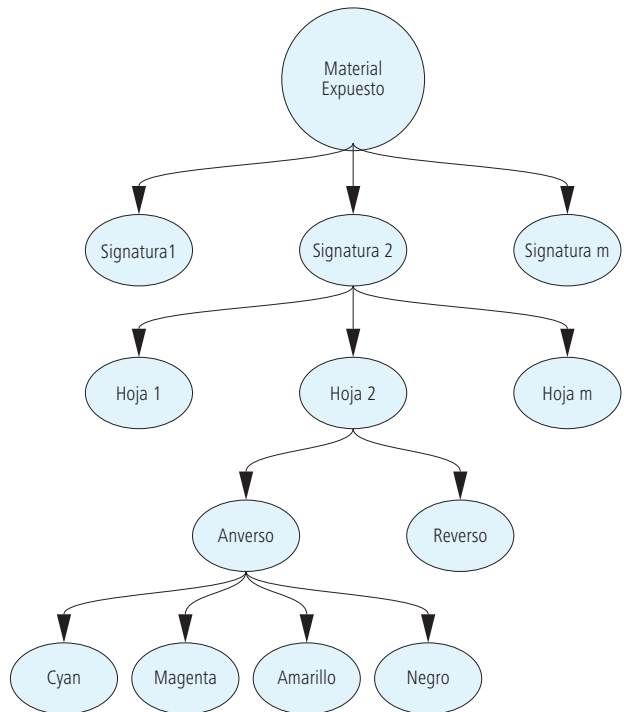
En nuestro caso, las hojas de impresión se dividen en 2 categorías; en buenos y malos como los actores de una película de Hollywood.

Todos los subelementos de los componentes padres y los dos componentes hijo adoptan los atributos parentales.

En el último ejemplo, la clasificación era de nivel único, pero en otros casos es necesario dividir en varios niveles.

El total de planchas necesarias para realizar el trabajo, como muestra la figura 6.15, se ha dividido en 4 niveles.

Figura 6.15
Recursos Componente del ejemplo Planchas



```

<ExposedMedia Class="Handling" ID="_4718" PartIDKeys="SignatureName SheetName
Side Separation" Status="Unavailable">
  <ExposedMedia SignatureName="Signature1">
    <ExposedMedia SheetName=" Sheet1" Status="Unavailable">
      <ExposedMedia Side="Front">
        <ExposedMedia Separation="Cyan" />
        <ExposedMedia Separation="Magenta" />
        <ExposedMedia Separation="Yellow" />
        <ExposedMedia Separation="Black" />
      </ExposedMedia>
      <ExposedMedia Side="Back">
        <ExposedMedia Separation="Black" />
      </ExposedMedia>
    </ExposedMedia>
  </ExposedMedia>
  ...
</ExposedMedia>
  ...
</ExposedMedia>

```

Figura 6.16 *Planchas Expuestas (ExposedMedia)* se pueden dividir en Nombre de Signatura (*SignatureName*), cada signatura en *Pliegos (Sheet)*, cada pliego en *Anverso (Front)* y *Reverso (Back)* y cada *Cara (Side)* en 4 separaciones.

En la figura 6.16 podemos ver la división de *ExposedMedia*, la lista dividida de recursos en color verde *PartIDKey*, *SignatureName* *SheetName* *Side Separation*, se describe exactamente igual que el diagrama de la figura 6.15. La secuencia de *PartIDKey* no se puede cambiar y siempre debe ir de “arriba abajo”.

La regla es la siguiente: cuanto más a la izquierda está un elemento, más cerca se encuentra del elemento raíz. Debemos respetar esta jerarquía al anidar recursos divididos.

En el ejemplo de la figura 6.16, la cara es CMYK y el reverso negro.

6.4 Cajas Grises y Procesos Combinados

Las razones para combinar procesos en una nueva estructura son:

- Al inicio de la producción no se concretaron los procesos. Considerar en este momento la producción completa es muy útil. Si no es posible, debemos declarar los procesos intermedios y los demás se definirán durante la producción.
- Si un programa o máquina ejecuta procesos sucesivos sin intervención no es necesario definir los procesos intermedios. P.E. Una imprenta digital con procesos integrados: interpretación, rasterizado, tramado, impresión digital,

grapado y corte. Otros procesos, como *PreviewGeneration* e *InkZoneCalculation*, pueden ser procesos combinados de preimpresión para impresión offset (sección 10.1).

El primer caso nos traslada al concepto de *Caja Gris* y el segundo al de proceso combinado.

CajaGris (GrayBox) es un nodo que agrupa procesos sin especificar más detalles. Estos se van añadiendo durante la producción, La *Caja Gris (GrayBox)* no puede aparecer como un proceso real y cuando todos los recursos y procesos están disponibles se cierra.

Normalmente, un MIS crea Cajas Grises porque en ese momento no se pueden especificar todos los parámetros de producción, por ejemplo, el MIS puede determinar las planchas que hacen falta y en que formato se producirán pero los datos de fabricación no son necesarios para realizar el presupuesto.

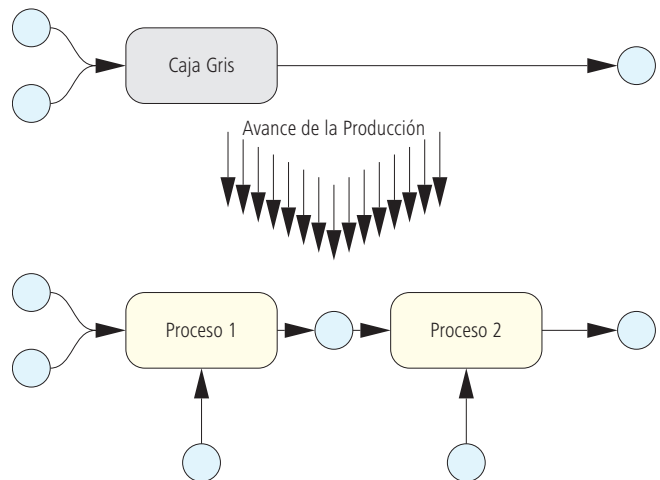
Por ejemplo para calcular el valor del reventado o el rasterizado se añaden valores por defecto obtenidos de los datos especificados en el propio trabajo o insertados por el operador en los módulos del flujo de trabajo.

En el esquema de la figura 6.17 vemos una *Caja Gris* que puede tener recursos de entrada y salida. Los recursos de salida están por especificar. En la figura 6.18 mostramos un ejemplo de *Caja Gris* que especifica la producción de planchas para impresión offset. Este caso está compuesto por 5 recursos de entrada y un recurso de salida, pero en otros casos pueden ser más.

Vamos a describir brevemente los recursos listados. En próximos capítulos los desarrollaremos en profundidad:

Figura 6.17
Resolución de Caja Gris durante la producción

- *Lista de Ejecución (RunList)*, es la descripción del conjunto de páginas de contenido ordenadas. Los datos de marcas también se necesitan para la imposición. En la figura 6.18, se denominan respectivamente *Documento* y *Marcas*.
- El Recurso *Maqueta (Layout)* describe las hojas o pliegos (hojas de imposición).



- *Control de Color (ColorantControl)* define los ajustes de color (espacios de color, colores, etc).
- *Sustrato (Media)* describe las características de las planchas de impresión (tipo, tamaño, nombre de producto, etc).
- *Material Expuesto (ExposedMedia)* en este caso, son las planchas expuestas.

Ahora repasemos el código de la figura 6.19 necesario para la generación de las planchas. Los nodos de proceso de grupo tipo *Caja Gris* siempre están compuestos por recursos y atributos *Type*. El valor de este atributo es *ProcessGroup* y el valor de los recursos *Type* es la lista de procesos contenidos en la *Caja Gris*. Podemos observar que *RIPing* no es un proceso en sí mismo, como ocurría en ejemplos anteriores, forma parte del nodo de proceso de grupo.

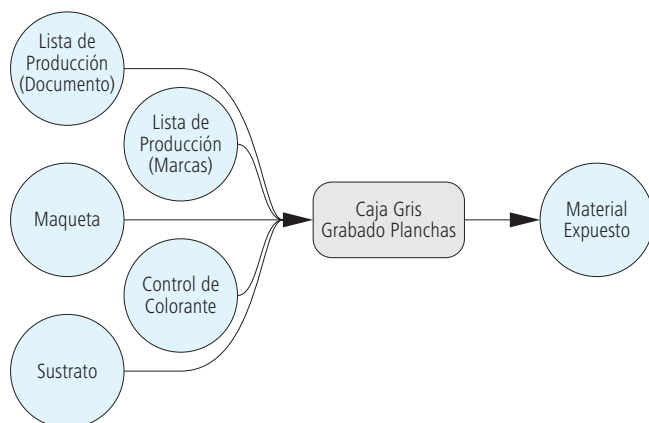


Figura 6.18 (arriba)
Ejemplo de Caja Gris para
realizar planchas de offset
Figura 6.19 (abajo)

Existen otros *nodos de proceso de grupo* que no tienen función de *Caja Gris*, en consecuencia, el atributo *Type ProcessGroup* carece de valor y se desestima. Este tipo de *nodos de proceso de grupo* contienen los procesos como nodos JDF.

Los procesos pueden estar agrupados y ser combinados. El resultado de los procesos combinados es otro proceso que contiene la lista de recursos y los procesos necesarios.

Podemos ver este caso en la figura 6.20, el valor del atributo *Type* es *Combined*. Los procesos bajo el valor *Types*, *Imposition*, *Interpreting* y *Rendering*, están listados al estilo de *Caja Gris*.

```
<JDF Type="ProcessGroup" Types="Imposition RIPing ImageSetting"
  DescriptiveName="GB PlateMaking" ...>
...
  <ResourceLinkPool>
    <RunListLink ProcessUsage="Document" Usage="Input" ... />
    <RunListLink ProcessUsage="Marks" Usage="Input" ... />
    <LayoutLink Usage="Input" ... />
    <ColorantControlLink Usage="Input" ... />
    <MediaLink Usage="Input" ... />
    <ExposedMediaLink Usage="Output" ... />
  </JDF>
```

```

<JDF Type="Combined" Types="Imposition Interpreting Rendering"...>
  <ResourcePool>
    <RunList ID="_100"... />
    <Layout ID="_200"... />
    <InterpretingParams ID="_300" ... />
    <RenderingParams ID="_400" ... />
    ...
  </ResourcePool>
  <ResourceLinkPool>
    <RunListLink CombinedProcessIndex="0" Usage="Input" rRef="_100" />
    <LayoutLink CombinedProcessIndex="0" Usage="Input" rRef="_200" />
    <InterpretingParamsLink CombinedProcessIndex="1" Usage="Input"
      rRef="_300" />
    <RenderingParamsLink CombinedProcessIndex="2" Usage="Input"
      rRef="_400" />
    ...
  </ResourceLinkPool>
</JDF>

```

Los conceptos proceso combinado y proceso de grupo son muy similares. La diferencia es que un proceso combinado se realiza en un solo dispositivo y el proceso de grupo se ejecuta en varios. Por esta razón, en procesos combinados no es necesario listar los recursos que describen los procesos intermedios.

Figura 6.20
Ejemplo de Proceso
Combinado para realizar
planchas de offset

El único atributo nuevo que podemos detectar en la figura 6.20 es *CombinedProcessIndex*. Este valor indica la relación entre recursos y procesos especificados bajo *Type*. Los procesos se cuentan desde cero; el primero de la lista *Imposition* tiene el número 0, el segundo *Interpreting* el 1 y el tercero *Rendering* el 2. Consecuentemente, los recursos de la *Lista de Producción* (*RunList*) y *Maqueta* (*Layout*) son recursos de entrada para *Interpreting*, mientras que el recurso de entrada para *Rendering* es *RenderingParams*.

6.5 Arquitectura del Flujo de Trabajo JDF

Ahora que ya hemos visto las bases para construir un documento JDF, examinaremos como se inserta en los módulos del flujo de trabajo.

La forma más simple es la transmisión secuencial de información, como se muestra en la figura 6.21. Este modelo no es muy realista y se utiliza únicamente en entornos de 2 o 3 dispositivos que intercambian información JDF, por ejemplo un programa de montaje o imposición que genera un documento JDF.

El modelo de la figura 6.22 es más efectivo; una central de datos accede a los archivos JDF mediante módulos incluidos en el flujo de trabajo.

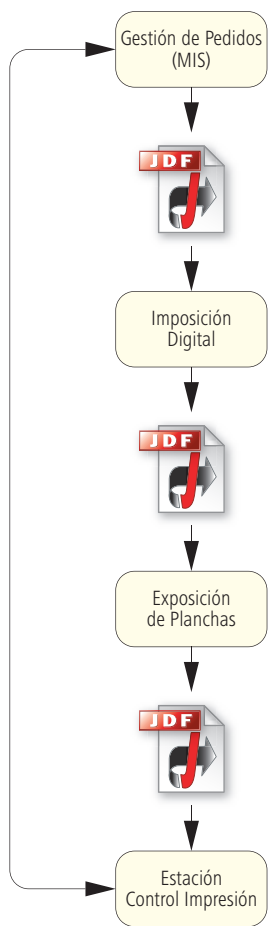


Figura 6.21
Transmisión secuencial

Este servidor puede ser una estructura de carpetas o una base de datos. Los datos JDF, como es lógico, no pueden estar a disposición de los distintos motores del flujo de trabajo, debe haber un software que administre la consistencia de los datos a un nivel superior. Todos estos datos se almacenan en un servidor central simplificando la creación de copias de seguridad, esta función resulta más complicada con el método secuencial.

La figura 6.22 sugiere que la administración JDF es parte del software de producción. Como vemos en la figura 6.23, también existe un modelo de MIS con gestor centralizado de datos.

En este caso no es necesario generar un JDF por pedido porque cada uno de los módulos del flujo de trabajo puede generar una nueva versión del documento JDF, lo cual por lo menos en teoría, nos facilita volver a un estado anterior del documento en caso de fallo,

No obstante, aclaremos para impedir equívocos que, incluso en entornos JDF, hay datos de producción que no se almacenan en dicho formato. Todos los datos para valorar la producción se almacenan en algún lugar del MIS. De hecho, el sistema continúa operando con todos los datos almacenados en sus bases de datos y tablas de valoración, pero por ejemplo, los costes de los equipos y operaciones no se integran en el JDF. Como ocurre en preimpresión, impresión y acabado, en sistemas JDF/JMF conviven varios tipos de interfaz externos.

La figura 6.24 muestra el modelo de organización jerárquica sugerido por CIP4. Este modelo se ocupa más del funcionamiento del software con JDF que del almacenamiento de los datos en sí.

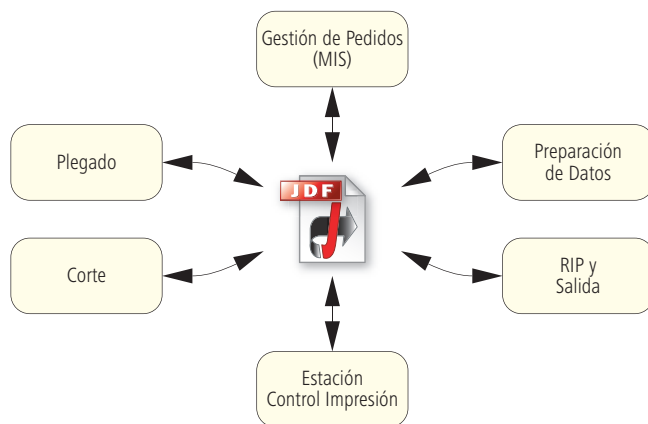


Figura 6.22
Administración central

A continuación mostramos la terminología asociada al tema:

- Un *Agente (Agent)* crea documentos y nodos JDF y modifica estos últimos. El Sistema de Gestión (MIS) adopta el rol de *Agente* habitualmente.
- Un *Controlador (Controller)* distribuye mensajes y documentos JDF a los dispositivos y recoge mensajes de los dispositivos que devuelve al *Agente*.
- Un *Dispositivo (Device)* interpreta nodos JDF e inicia las acciones correspondientes en máquina. La comunicación Dispositivo/Máquina no es JDF/JMF, es propia del proveedor. En argot informático se conoce como "Controlador JDF".
- La *Máquina (Machine)* procesa sin crear ni entender JDF/JMF.

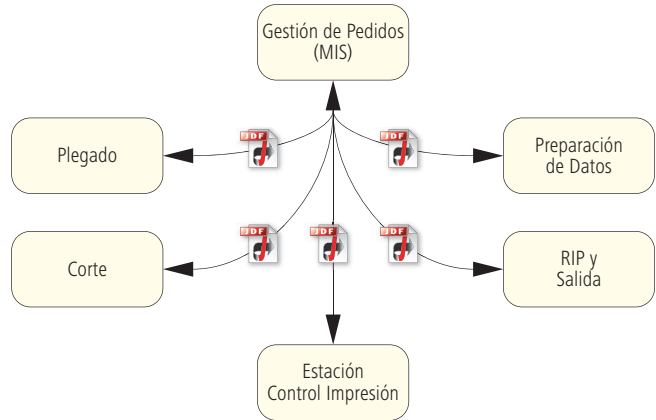


Figura 6.23
El MIS es el responsable de la comunicación JDF

La división no siempre está clara. Hay controladores/dispositivos que tienen facultades de *Agente* y pueden producir o modificar nodos JDF. Además pueden ser *Controlador* y *Agente* a la vez, como ocurre con ciertos Sistemas de Gestión de Pedidos. En consecuencia el *Agente* (MIS) también es un *Controlador* universal. Esta arquitectura JDF tiene 3 niveles.

La división en 3 Controladores de la figura 6.24 solo es un ejemplo, en realidad sería un solo *Controlador*. Podemos decir como conclusión que un *Controlador* puede gestionar *Controladores* subsidiarios o dicho de otro modo, puede trabajar de forma recurrente.

Un *Controlador* puede transmitir datos simultáneos utilizando los siguientes métodos:

- Un *Controlador* envía el trabajo JDF completo a todos los dispositivos por turnos.
- El *Controlador* determina los procesos que puede realizar cada dispositivo mediante JMF y envía al dispositivo adecuado la parte respectiva del trabajo JDF.
- De igual modo que en el último punto, el *Controlador* solo se conecta con las propiedades particulares del dispositivo.

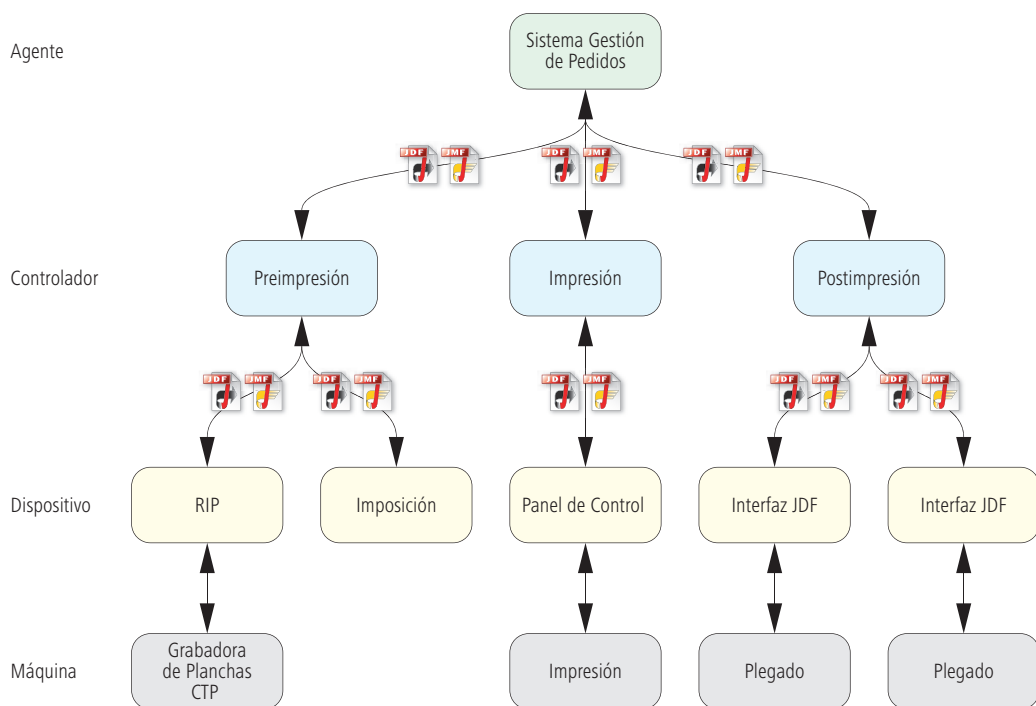


Figura 6.24
Organización jerárquica de
comunicación JDF/JMF

Los 2 últimos puntos necesitan sus partes correspondientes de los nodos JDF. En la siguiente sección describimos la tecnología utilizada para realizar la separación.

La combinación de equipos y dispositivos se denomina "motores de flujo de trabajo" o "procesadores de órdenes de trabajo". A primera vista se nos plantea la cuestión de qué procesos deben ser automatizados para que la instalación JDF sea rentable porque los modelos presentados parecen bastante estáticos y complejos, pero de hecho los dispositivos JDF pueden iniciar una máquina automáticamente. A continuación exponemos las condiciones necesarias para poner en marcha estos procesos:

- Un proceso descrito en JDF es ejecutable por la máquina.
- El estado del proceso es *Preparado* o *En Espera*.

Figura 6.25
Recurso NodeInfo

```
<ResourcePool>
  <NodeInfo FirstStart="2008-08-01T00:00:00+01:00"
    LastStart="2008-08-02T08:08:00+01:00"
    LastEnd="2008-08-02T15:00:00+01:00"... />
  ...
</ResourcePool>
```

- Todos los recursos tienen el estado *Disponible*.
- La hora actual está dentro del plazo de producción establecido.

El plazo de producción se registra en un recurso propio denominado *NodeInfo*, también suele incluir comentarios sobre la ejecución o los motivos de un posible retraso o los datos de responsable del proceso. El proceso del ejemplo en la figura 6.25 debe iniciarse después de la medianoche del 01/08/2008 y no se debe demorar más allá de las 8 de la mañana del día 02/08/2008. Debe concluir como muy tarde a las 15 hrs. del mismo día.

6.6 Separar y Combinar

Los sistemas de flujo de trabajo pueden funcionar en paralelo, pero cada dispositivo JDF debe disponer de la información necesaria. Además, es más práctico suministrar al dispositivo la información que precisa, que todo el documento JDF. Para poder realizarlo se deben duplicar y enviar secciones del código JDF a los demás dispositivos.

Cada dispositivo puede agregar información JDF a los archivos y es necesario disponer del registro de producción, por lo tanto debería existir un nuevo documento JDF disponible en algún lugar del sistema desde donde fuera posible incorporar los datos copiados, el JDF modificado y el documento original. Para que todo esto sea posible debemos considerar ciertos aspectos. Algunas secciones del código JDF son inamovibles y se deben establecer los permisos para leer (*read-only*) o leer y escribir (*read-write*). En resumen, debemos garantizar la consistencia de los datos.

En argot JDF, este proceso se denomina "Separar y Combinar" (*Spawn&Merge*). Preferimos llamarlo "separar", en vez de "desglosar", aunque en realidad es un proceso de copia.

La figura 6.26 muestra gráficamente las diferentes posibilidades. En la (1) *Estado Inicial* se separa un nodo JDF del documento original y genera un nuevo documento *Separar/Desglosar* (2). El nodo separado se modifica en el punto *Producción Avanzada* (3), finalmente, se combina con el árbol JDF original en el punto *Reunir/Combinar* (4). Al mismo tiempo, el nodo individual puede generar otros nodos secundarios.

Los recursos pueden ser divididos pero siempre debe quedar consistencia de los recursos divididos en el documento original. En el *Fondo de Auditoría (AuditPool)* se registran todas las acciones de *Separación y Combinado*.

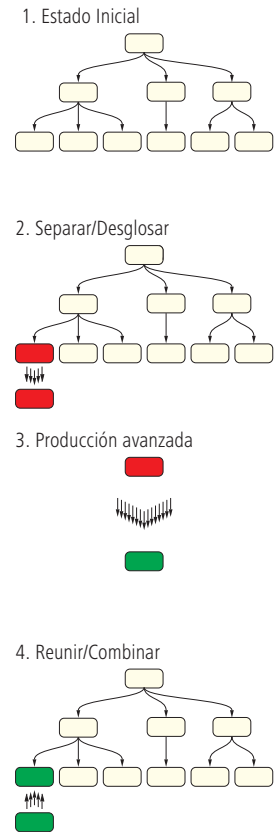


Figura 6.26 "Separar" y "Combinar" Procesos de separación y reunificación, respectivamente.

```

<JDF DescriptiveName="Total product" JobID="_001"...>
  <JDF ...DescriptiveName="Cover" ID="_002 " JobPartID="_003"
    Status="Waiting" Type="Product">
    ...
  </JDF>
</JDF>

```

Figura 6.27
Archivo original
antes de la separación

A continuación, desglosamos estas operaciones con un ejemplo. La figura 6.27 muestra el código del trabajo JDF antes del proceso. En el paso siguiente se separa el subproducto "Cubierta" (para ser producido, por ejemplo, en otro lugar). En la figura 6.28 podemos ver el cambio del archivo maestro tras la separación: el *AuditPool*, incluido en el archivo separado, registra el subnodo *NewSpawnID*. Junto al identificador del nodo separado *jRefID* se puede ver su URL y el atributo separado *rRefsROCopied* (R_1 a R_9). *RO* significa *Recurso de Lectura (Read-Only)*, por lo tanto el proceso encargado de la parte separada no se puede alterar, solo se puede leer. Además, aún podemos ver las piezas separadas del árbol JDF en el archivo original – en este caso es el nodo JDF *Cubierta (Cover)* –. La figura 6.29 muestra los datos JDF separados de dicha sección. Los identificadores *JobID* y *JobPartID* permanecen sin cambios, solo se define *SpawnID* (es el mismo valor del atributo *NewSpawnID*). El elemento *Antepasado (Ancestor)* reúne la localización del archivo y la identificación de la tarea (*JobID*), añadidos al JDF original.

Figura 6.28 (arriba) Archivo
Original tras la separación
Figura 6.29 (abajo)
Archivo JDF separado

```

<JDF DescriptiveName="Total product" JobID="_001"...>
  <AuditPool>
    <Spawned NewSpawnID="_004" Status="Waiting"
      TimeStamp="2008-10-25T09:38:09+02:00"
      URL=file:/C:/GetrennteDatei.jdf
      jRef="002" rRefsROCopied="R1 R2 R3 R4 R5 R6 R7 R8 R9" />
  </AuditPool>
  <JDF DescriptiveName="Cover" ID="_002" JobPartID="_003"
    Status="Spawned" Type="Product">
    ...
  </JDF>
</JDF>

```

```

<JDF ... DescriptiveName="Total product" ID="_002" JobID="_001"
JobPartID="_003"
  SpawnID="_004" Status="Waiting" Type="Product">
</JDF>
...
<AncestorPool>
  <Ancestor FileName="file:/C:/Originaldatei.jdf" JobID="_001"... />
  ...
</Ancestor>
</AncestorPool>
</JDF>

```



```

<JDF DescriptiveName="Total product" JobID="_001"...>
  <AuditPool>
  ...
    <Spawned NewSpawnID="_004" Status="Waiting"
      TimeStamp="2008-10-25T09:38:09+02:00" URL="file:/C:/Test.jdf"
      jRef="002" rRefsROCopied="R1 R2 R3 R4 R5 R6 R7 R8 R9 />
    <Merged ... MergeID="_004" TimeStamp="2008-10-25T09:41:01+02:00"
      URL="file:/C:/GetrennteDatei.jdf" jRef="002" />
  </AuditPool>
  ...
  <JDF DescriptiveName="Cover" ID="_002" JobPartID="_003"
    Status="Waiting" Type="Product"...>
  ...
  </JDF>
</JDF>

```

Figura 6.30
Archivo JDF tras
la reunificación

Esta información es necesaria para reincorporar los datos. En su estado actual, el archivo principal y el separado se podrían gestionar de manera independiente, siempre que este último tuviera derechos de escritura. Por último se fusionan los 2 árboles JDF. El resultado de la figura 6.30 es exactamente igual que antes de la separación excepto la entrada *AuditPool*, donde se describe que una sección se separó y más tarde se volvió a combinar.

El proceso completo es recurrente, lo que significa que los elementos JDF separados pueden volver a separarse de nuevo.

6.7 Especificación de Conformidad e Interoperabilidad (ICS)

Un interfaz siempre puede resultar problemático. Por ejemplo, si una agencia de publicidad solicita un producto a una imprenta, es posible que la agencia desconozca la forma correcta de enviar los datos. En orden inverso, la imprenta desconoce los conocimientos de la agencia al respecto, esta situación provoca malentendidos y frustraciones. Aunque hablan el mismo idioma su comunicación no es satisfactoria. La solución es utilizar los estándares PDF/X [23] o PDF/XPlus [19] para la producción de datos de contenido, aun- que su uso limita ciertas funciones del formato PDF.

En el flujo de trabajo JDF también se producen estos problemas de interfaz: cuando un *Agente*, *Controlador* o *Dispositivo* genera una estructura JDF, se espera que el receptor pueda usarla. En el caso contrario, el receptor espera recibir los datos JDF de cierta manera y posiblemente no es así.

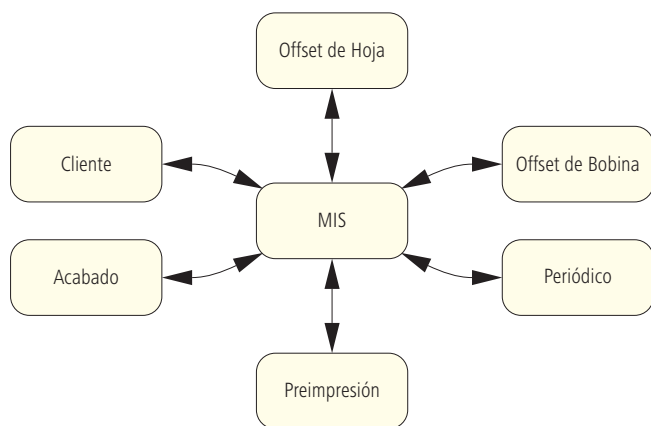
Las **Especificaciones de Conformidad de Interoperabilidad (ICS)** ayudan a resolver estas situaciones. Su función es limitar las posibilidades que JDF/JMF proporciona a ciertos componentes de software.

En este tipo de problemas siempre se plantean dos roles: el generador de información y el consumidor. Para afrontar esta cuestión los conceptos de *Agente*, *Controlador* y *Dispositivo* no son suficientes, necesitamos dos nuevos:

- El *Administrador*. Envía los documentos JDF y mensajes JMF opcionales a *Trabajador*, *Controlador* o *Dispositivo*. Su función secundaria es recibir y leer mensajes y documentos JDF de los *Trabajadores*.
- El *Trabajador*. Recibe documentos JDF del *Administrador* (*Agente* o *Controlador*) y mensajes JMF opcionales. Su función secundaria es devolver información JDF/JMF.

Administrador (Manager) y *Trabajador (Worker)* son las denominaciones utilizadas por las especificaciones ICS y a partir de ahora son las que usaremos nosotros. El *Administrador* y el *Trabajador* leen y escriben datos JDF/JMF. Un ejemplo, el *Administrador* envía al *Trabajador* un proceso JDF, este lo realiza y devuelve la información sobre recursos usados, consumidos y generados, durante la producción proceso o una vez ha finalizado. Si quisiéramos enfatizar los roles de escritor/modificador JDF/JMF deberíamos denominarlos: *Productor (Producer)* al *Administrador* y *Consumidor (Consumer)* al *Trabajador*.

Figura 6.31
Interfaces MIS descritos en
los documentos ICS



Las especificaciones ICS describen diferentes entornos gráficos de comunicación (Interfaces) complementarios entre sí; estos se construyen unos sobre otros o al menos en parte. En la figura

6.31 vemos todos los interfaces MIS basados en la versión JDF 1.3 y definidos en las especificaciones ICS. En la figura 6.32 nos ayuda a comprender su organización jerárquica. *MIS a Acabado ICS* se construye sobre *Base ICS*, *MIS ICS* y *JMF ICS*. *Impresión Digital de Oficina ICS*, solo está basada en *Base ICS*.

Los interfaces están estructurados en niveles jerárquicos.

Ahora vamos a explicar la especificación *Base ICS* concretamente. Trataremos las demás en próximos capítulos.

Hay 3 niveles definidos: Nivel 0, Nivel 1 y Nivel 2. El Nivel 1 integra funcionalidad de Nivel 0 y el Nivel 2 de Nivel 1. Los niveles propuestos de *Base ICS* también se pueden utilizar en las demás especificaciones ICS.

El interfaz cumple el Nivel 0 de *Base ICS* si el *Administrador* transmite archivos JDF al *Trabajador* mediante una carpeta activa y este puede leerlos. El Nivel 1 se cumple cuando el *Trabajador* devuelve los archivos JDF al *Administrador* y este puede leerlos. El Nivel 2 necesita dos requerimientos adicionales:

- Para implementar el WMS se necesita transmitir datos simultáneamente, como documentos JDF, mensajes JMF, datos de impresión PDF, perfiles de Preflight e ICC y previos de imagen. Pueden llegar por correo electrónico agrupados en paquetes **MIME (Multipurpose Internet Mail Extensions)** (Extensión Multifunción para Correo por Internet). El Nivel 2 permite al *Administrador* recibir paquetes MIME y al *Trabajador* interpretarlos (figura 6.33).
- En general, los archivos (p.ej. PDF) se asocian mediante referencias contenidas en los documentos JDF (ruta a los archivos en el servidor local).

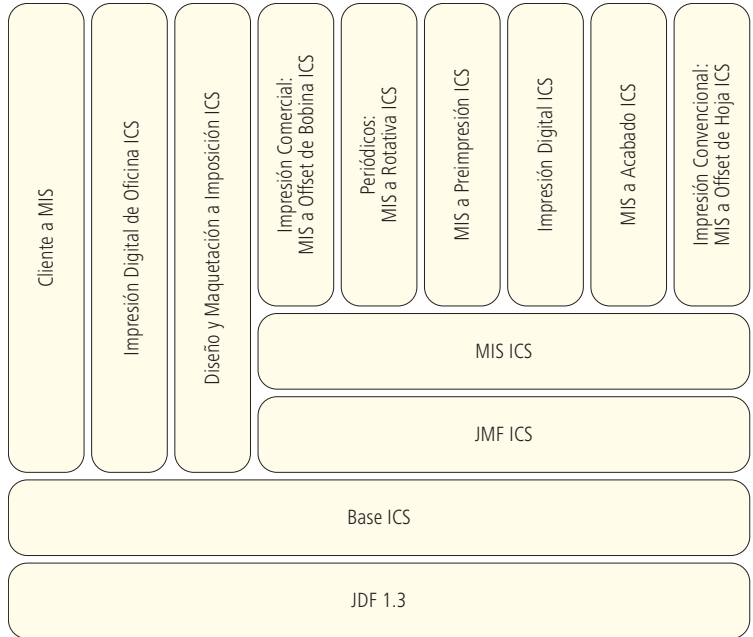


Figura 6.32 Las especificaciones ICS se desarrollan, en parte, unas basadas en las otras

Figura 6.33 Diferentes niveles en Base ICS

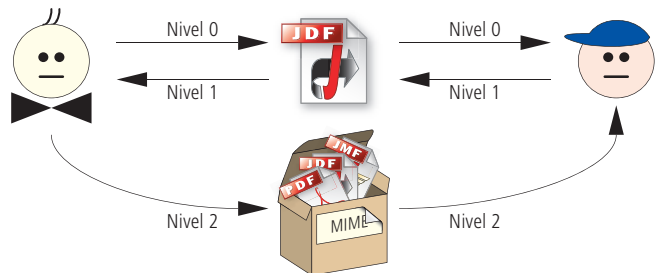


Figura 6.34
Referencia a los Datos de
Contenido vía URL

En la figura 6.34, la secuencia de página *RunList* define que es un archivo PDF (*LayoutElement*) e indica el nombre y ubicación mediante la ruta (*FileSpec*). El valor "0 ~ - 1" del atributo *Pages* concreta que se usan todas las páginas. Además, el Nivel 2 exige a ambos, *Administrador* y *Trabajador*, realizar la entrega y recogida de los archivos a través de HTTP.

```
<ResourcePool>
  <RunList ID="_4719,, Pages="0~-1" Class="Parameter" Status="Available">
    <LayoutElement>
      <FileSpec URL=file://Workflowserver/Pagefiles/Frisch-Advertising.pdf
    />
    </LayoutElement>
    ...
  </RunList>
  ...
</ResourcePool>
```

Aparte de los niveles descritos, la *Base ICS* trata otras cuestiones técnicas como el número de caracteres del identificador *ID*. También resultan interesantes los requisitos de los nodos JDF, sus atributos y subelementos. En la configuración de los estados, por ejemplo, está claro que el *Administrador identifica ID* la tarea en el elemento raíz y el *Trabajador* debe leerla, pero ¿quién está obligado a insertar los datos en el *AuditPool*? o ¿el responsable debería ser opcional?

En resumen, se puede decir que las especificaciones ICS no son únicamente herramientas para desarrolladores JDF, también sirven para probar la integración de los distintos componentes. Si por cualquier causa falla la comunicación JDF, lo primero que debemos examinar es que los componentes involucrados cumpan las reglas del documento ICS.

Ejercicios

Descargue algunos archivos JDF de ejemplo y *JDF Editor* de la página web de CIP4.org. Abra los archivos y examine la estructura de los nodos JDF y sus recursos.

A continuación, renombre los archivos de *.jdf a *.xml y abra estos últimos en un navegador para analizar el código XML directamente.

Separe un subnodo JDF con la ayuda del *JDF Editor* y, a continuación, vuelva a combinar el subnodo separado. Seleccione la opción *Spawn & Merge* del menú *herramientas*. Analice el estado intermedio en un navegador.

7 Formato de Mensajería de Trabajo JMF

En el capítulo anterior ya vimos la transmisión de datos entre los diferentes Agentes, Controladores y Dispositivos JDF. En el caso más simple, un Productor JDF escribe datos JDF en una carpeta controlada a intervalos temporales por un Consumidor JDF. Ciertamente, este interfaz unidireccional de transferencia de archivos es muy estático y la interacción dinámica entre las partes del flujo de trabajo es prácticamente nula. Sobre todo al inicio la recopilación de datos y el seguimiento de los trabajos es muy importante para la gestión del sistema: comprobar si se producen cambios, analizar el consumo de material, detectar los errores de forma centralizada y valorar la eficiencia global de las máquinas.

El **Formato de Mensajería de Trabajo JMF** se introdujo junto a JDF con este objetivo. Aunque JMF aporta un gran valor añadido, no se suele implementar en todos los flujos de trabajo JDF. Para comprender su importancia vamos a analizar el ejemplo de un MIS que requiere Nivel 2 para disponer de mensajes JMF.

Conocer la tecnología es importante para entender el funcionamiento general del flujo de trabajo JDF/JMF, aunque no lo es tanto a nivel usuario porque los datos JMF se envían por protocolos HTTP y no están físicamente en el fichero, mientras que los datos contenidos en los archivos JDF son más fáciles de analizar para solventar posibles problemas. Existen programas que permiten leer y rastrear paquetes HTTP, se denominan **HTTP sniffer** pero por la red circula una gran cantidad de datos y analizarlos no es divertido. De cualquier forma pondremos más énfasis en JDF que en JMF por que es lo que nos interesa.

7.1 Modelos de Comunicación

Antes de entrar en la estructura de los protocolos de mensajes JMF, veamos los modelos de comunicación entre módulos de software, utilizados habitualmente por la industria gráfica:

- Transferencia manual o vía *carpetas activas* (*hot folders*).
- Transferencia de datos a través de protocolos de impresión.
- Interfaces de bases de datos.
- Comunicación entre procesos.
- Servicios Web.

- Protocolos de mensajería.

Aunque la transferencia de archivos mediante carpetas activas tiene la ventaja que supone la simplicidad de gestión para el usuario, especialmente para administradores del sistema, también tiene algunos inconvenientes:

- Cuando se rompe la cadena de una carpeta activa, esta no envía mensaje de fallo (por ejemplo el error de un equipo).
- El Productor no recibe la confirmación de que sus datos se leyeron correctamente y pueden interpretarse.
- En caso de error en la ejecución del trabajo no existe monitorización del sistema que detecte las causas.
- La comunicación en las carpetas activas es lenta y no es adecuada para la recogida de datos de planta o la coordinación de módulos de información bidireccional.

Todavía existen RIPs que usan carpetas activas para recibir los datos de impresión. Habitualmente, los diversos ajustes de RIP (resolución, lineatura, etc) se asocian a estas. Pero puede resultar confuso configurar tantas carpetas activas con diferentes ajustes de RIP. En un entorno JDF no existe este problema, porque la información está contenida en los metadatos y las variables las introduce el RIP, por lo tanto, las carpetas activas solo son el canal de comunicación.

Otra forma de distribuir archivos a los sistemas RIP son los protocolos de impresión. Si se hace vía menú de impresión, el sistema debe realizar ciertos ajustes para imprimir un trabajo y el método no es adecuado para JDF. La transmisión de JDF mediante bases de datos tampoco es posible porque no existen métodos de acceso. Con la comunicación de procesos realizados por distintos fabricantes ocurre lo mismo porque no utilizan código abierto.

En la comunicación PJTF de los distintos procesadores de órdenes de trabajo encontramos todas estas técnicas.

Los servicios Web basados en XML [45] definen el intercambio de mensajes entre aplicaciones en la red. Por ejemplo, los sistemas de las agencias de viajes se comunican vía web con los hoteles para realizar consultas y reservas. Los módulos de un sistema gestor de flujos de trabajo pueden comunicarse de la misma manera.

Los servicios Web necesitan protocolos de mensajería, por ejemplo SOAP **Simple Object Access Protocol** de W3C. De hecho, es el protocolo que usan los sistemas de gestores de flujo de trabajo.

Los servicios SOAP utilizados para el intercambio de mensajes XML también transportan JMF. HTTP es el sistema utilizado para el envío.

Figura 7.1
Mensaje JMF

Como podemos ver en la figura 7.1, los mensajes JMF son archivos XML donde el elemento raíz lleva por nombre JMF. En los subelementos del elemento raíz JMF es donde se codifican realmente los mensajes. El mensaje, propiamente dicho, se envía a través de HTTP o por protocolo HTTPS utilizado habitualmente en internet para el transporte de páginas web. Con este método se puede construir rápidamente una comunicación de dos vías. Alternativamente, algunos mensajes JMF se insertan en carpetas activas como archivos que el receptor debe leer. Es el método más utilizado con JDF (figura 7.2).

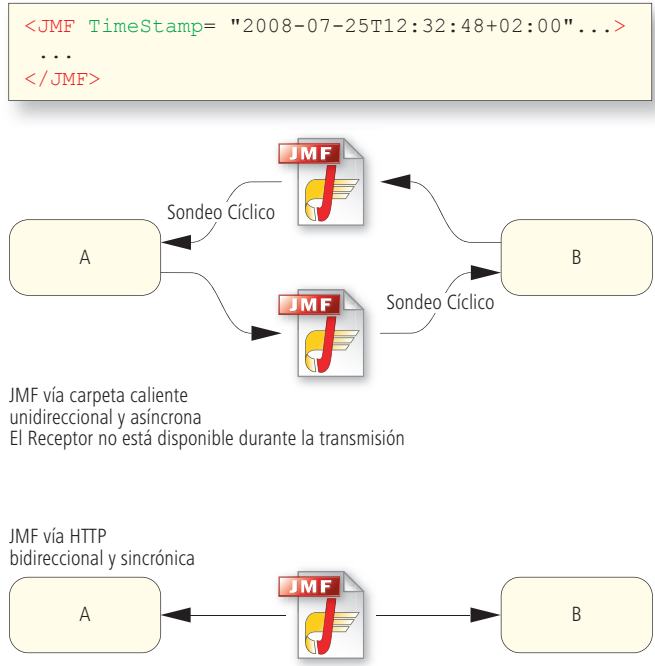


Figura 7.2
Diferentes tipos de comunicación JMF

7.2 Familias JMF

Actualmente con JDF ¿qué contenidos se pueden transportar?. Exponemos algunos ejemplos a continuación:

- Activación de dispositivos.
- Dispositivo y estado de trabajo.
- Control de colas y entradas.
- Cumplimiento de etapas.

La divulgación del estado de las órdenes y dispositivos son de especial interés por los diferentes beneficios que conlleva. El seguimiento del trabajo es posible con JDF e incluso el análisis del coste final. Además, con dicha comunicación es más fácil la supervisión centralizada de los dispositivos. En la práctica, es relativamente sencillo generar y enviar mensajes sobre el progreso de la producción y las condiciones del dispositivo. Lo que resulta más complejo es filtrar tantos datos para obtener detalles significativos.

El coste final no se calcula en la base de datos JMF por esta razón, en su lugar se utilizan los elementos *AuditPool* que presentamos anteriormente en la sección 6.2. Las entradas *AuditPool* de los dispositivos una vez completan sus procesos se transportan en el propio JDF hasta el sistema central JDF o el MIS y el *Administrador* redacta la *Auditoría* de *Trabajadores* tras filtrar los mensajes JMF.

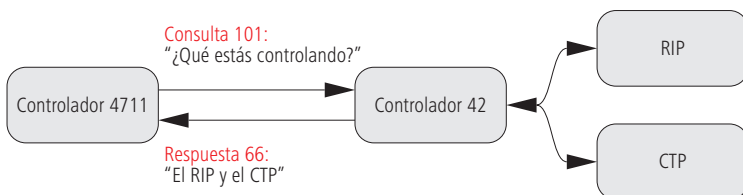
Los mensajes JMF se dividen en 6 categorías conocidas como familia JMF:

- *Consulta* JMF.
- *Comando* JMF.
- *Respuesta* JMF.
- *Reconocimiento* JMF.
- *Señal* JMF.
- *Registro* JMF.

La *Consulta JMF* al controlador o dispositivo JMF se realiza para obtener una información concreta y no cambia el estado del destinatario, al contrario de lo que ocurre con el *Comando JMF*. Este dispositivo/controlador devuelve una *Respuesta*. En la figura 7.3, podemos ver gráficamente una *Consulta* y la debida *Respuesta*. Cada controlador o dispositivo tiene su identificador ID, al igual que la consulta y su respuesta. De modo que cada *Respuesta* está asociada inequívocamente a su *Consulta*. En este caso, el controlador $ID = 4711$ desea saber que dispositivo administra el $ID = 42$, la respuesta es que controla un RIP y un CTP. Esta pregunta es importante para el hipotético método conectar y listo (Plug&Play JDF) de módulos JDF, aunque esta situación todavía está lejos de la realidad.

En la figura 7.4 podemos ver el código JMF correspondiente. Tal vez nos resulte obvio que la ID del receptor no se incluye ni en la *Consulta* ni en la *Respuesta* del código XML. En la *Consulta* no está el $ID = 42$ ni $ID = 4711$ en la *Respuesta*. Entonces ¿cómo llega el mensaje a sus destinatarios? La respuesta es que el receptor se define como las carpetas activas, mediante el protocolo HTTP.

Figura 7.3
Consulta y Respuesta JMF



```

<JMF TimeStamp="..." SenderID="_4711">
  <Query Type="KnownDevices" ID="_101"/>
</JMF>

<JMF TimeStamp="..." SenderID="_42">
  <Response Type="KnownJDFServices" ID="_66" refID="_101"/>
  <DeviceList>
    <DeviceInfo DeviceStatus="Idle">
      <Device DeviceID="Rip" />
    </DeviceInfo>
    <DeviceInfo DeviceStatus="Running">
      <Device DeviceID="CtP"/>
    </DeviceInfo>
  </DeviceList>
  <Response>
</JMF>

```

En sentido figurado, el mensaje JMF se encuentra en un sobre etiquetado con la dirección correspondiente y se envía a través de HTTP. También podemos ver claramente en el código que la respuesta dirigida a la solicitud se establece en el atributo *refID*.

Figura 7.4
Respuesta a Consulta JMF

Todos los *Comandos* pueden cambiar las cosas, mensajes *JMF* incluidos. Una *Consulta* solo puede solicitar información sobre el estado de una cola pero un *Comando JMF* permite borrar entradas de dicha cola.

Antes de continuar debemos definir el concepto, resulta obvio que no es la cola del supermercado. Una Cola es una estructura reguladora de datos que normalmente usa el método *El primero que llega/Primero sale (First In/First Out)*. Esto también se puede aplicar al *Buffer* de un RIP, por ejemplo, donde los datos de trabajos JDF se almacenan temporalmente en un silo de memoria para un controlador o dispositivo.

Si hablamos de colas se suelen utilizar dos tipos de comandos JMF: los relativos a la cola o los referidos a cada entrada de una cola. El comando *Retener Cola (HoldQueue)* la detiene y no procesa más trabajos y el comando *Retener Entrada en Cola (HoldQueueEntry)* detiene un solo trabajo de la cola.

Debemos considerar que la funcionalidad de los flujos de trabajo sin ellos sería muy limitada, aunque puedan parecer poco útiles en principio. Por ejemplo, si un dispositivo puede comunicar al controlador que el trabajo no está en espera y ya está en ejecución, el usuario puede conocer el estado de la producción en su conjunto y utilizar la información en tiempo real para otros cometidos.

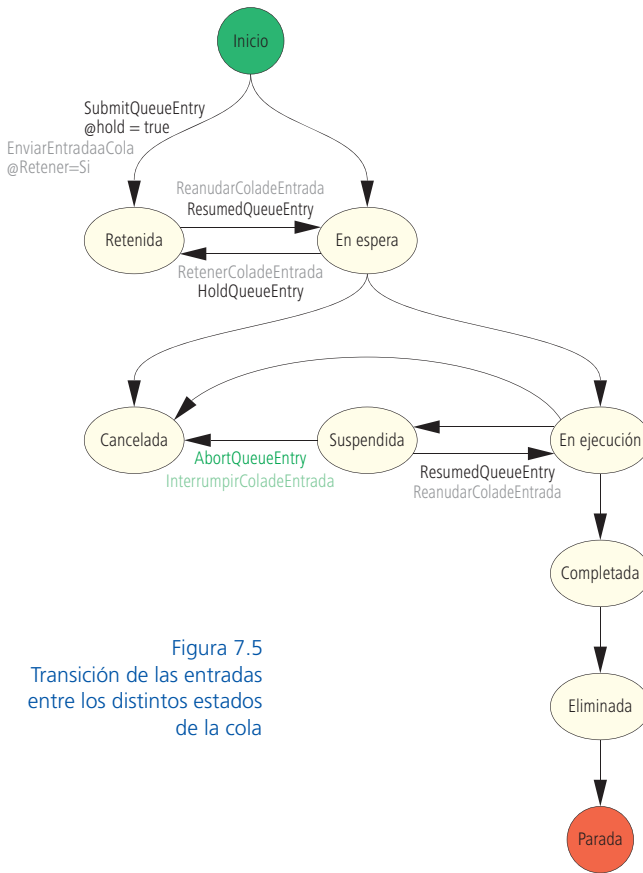


Figura 7.5
Transición de las entradas
entre los distintos estados
de la cola

Figura 7.6
Cancelar un trabajo.
Comando y respuesta
correspondiente

Las entradas de una cola tienen diferentes estados:

- *Espera (Waiting).*
- *Retenida (Held).*
- *Ejecución (Running).*
- *Detenida (Suspended).*
- *Cancelada (Aborted).*
- *Eliminada (Removed).*

Los estados corresponden a la cola y a los dispositivos como vemos en la figura 7.4. *Estado de los Dispositivos (DeviceStatus).*

La figura 7.5, muestra los estados y sus transiciones (algunos no aparecen para no sobrecargar el dibujo). También incluimos los comandos JMF causantes de las transiciones. Observe la diferencia entre *Retenida* y *Suspendida*, en ambos casos, los trabajos se envían para ser procesados más tarde, una vez desde *En Espera* y otra al salir del estado *En Ejecución*.

```
<JMF SenderID="Controller-1" TimeStamp="2008-08-13T10:05:32+01:00"
Version="1.3"...>
<Command ID="C1" Type="AbortQueueEntry">
  <QueueEntryDef QueueEntryID="job-4711" />
</Command>
</JMF>

<JMF SenderID="Device-1" TimeStamp="2008-08-13T10:05:33+01:00"
Version="1.3"...>
  <Response ID="R1" Type="AbortQueueEntry" refID="C1">
    <Queue DeviceID="Device-1" Status="Running">
      <QueueEntry JobID="job-4711" QueueEntryID="job-4711"
        Status="Aborted" />
    </Queue>
  </Response>
</JMF>
```

En la figura 7.6 vemos el ejemplo de un comando utilizado para *Cancelar* un trabajo (*AbortQueueEntry*) y su *Respuesta* (*Response*). El *Controlador 1* envía el comando *ID = "C1"*, referente al trabajo de la cola de entrada "*job-4711*". El receptor de este comando es el dispositivo *Device-1*; su comando de *Respuesta* se refiere a *C-1*. El trabajo cancelado todavía es una entrada en la cola, la cancelación solo se produce cuando el *Estado* sea igual a "*Aborted*".

Los trabajos JDF se pueden transferir como archivos a través de carpetas activas, como ya hemos comentado anteriormente. Otra posibilidad es que un dispositivo de comunicación o controlador gestione las transferencias y recoja los ficheros JDF. Para esta opción existen 2 posibilidades, una es enviarlo como un archivo normal y otra transferirlo mediante HTTP. La figura 7.7 muestra la transferencia del archivo.

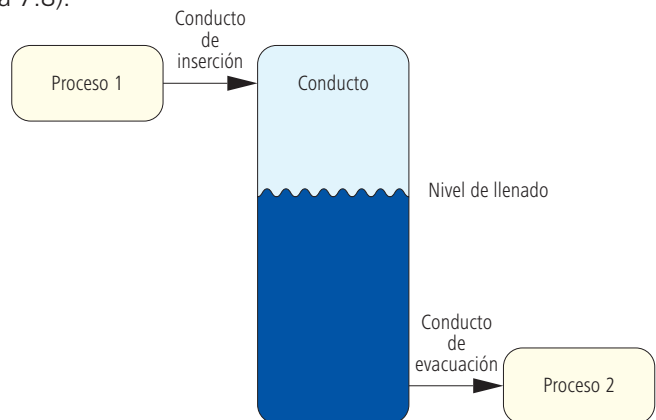
Figura 7.7
Entrada de cola individual

```
<Command ID="C2" Type="SubmitQueueEntry">
  <QueueSubmissionParams URL="File://HdM/Server1/Jobs/job1.jdf" />
</Command>
```

Otro ejemplo de comandos es la estructura tratada brevemente en la sección 2.6 "tubería" de datos. Es un tipo de producción en cadena donde el proceso P1 se inserta en un tubo y un segundo proceso P2 lo devuelve tras un lapso de tiempo. Lo especial de este método es que un recurso no necesita estar completamente en el tubo para ser leído. En el capítulo 2 mostrábamos un ejemplo; el proceso P1 era la formación de imágenes para realizar las planchas y el proceso P2 era la impresión. Ya se podía imprimir con el primer juego de planchas mientras se grababa el segundo. Las acciones para insertar recursos en el tubo *Conducto de Inserción* (*PipePush*) y para eliminarlos *Conducto de Evacuación* (*PipePull*) son comandos igualmente (Figura 7.8).

Figura 7.8
Almacenamiento y comandos JMF para el llenado y vaciado, basado en el concepto "tubería" tratado anteriormente.

Finalmente, hablaremos de los **Mensajes de Recursos**. Son solicitudes o comandos concernientes a recursos JDF. Los recursos: materiales, equipos o trabajos se solicitan (*ResourceQuery*) o modifican (*ResourceCommand*). Con los comandos, podemos actualizar recursos o enviar nuevas versiones del documento JDF.

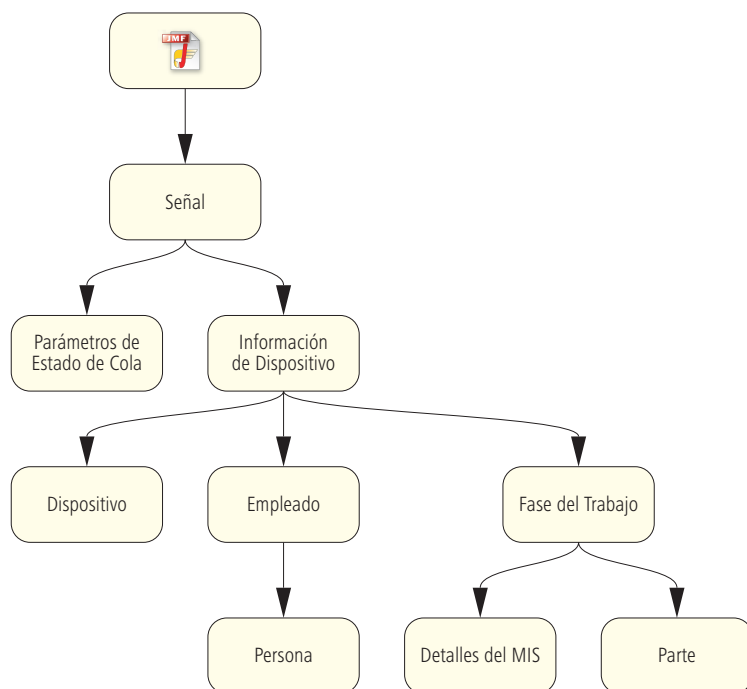


Por ejemplo, el controlador correspondiente envía un comando si se graba una plancha y el recurso JDF, o una sección del mismo, pasa al estado *Disponible*.

En JMF la respuesta a una Consulta o comentario tras el *Acuse de Recibo (Acknowledge)* se puede retrasar (asíncrono). El receptor envía una respuesta que comunica la recepción del comando (consulta) y confirma que el comando se ha realizado, la respuesta al comando se retrasa porque necesita cierto tiempo para su ejecución. En la vida real es al contrario: la orden se realiza inmediatamente y se responde más tarde.

Los mensajes unidireccionales sobre los cambios de estado de trabajos y máquinas se denominan señales. Un controlador notifica el tipo de señal que debe recibir, significa que puede “suscribirse” a las señales de diferentes maneras. La figura 7.9 muestra una estructura de señal JMF que informa del estado de una máquina offset. Los mensajes JMF son señales que suelen contener 2 subelementos. El elemento que define el trabajo es *StatusQuParams*. El elemento *DeviceInfo* informa sobre el *Estado Actual* del dispositivo y detalles como los datos de los contadores de producción y otros.

Figura 7.9
Señal JMF



Los subelementos *Device*, *Employee* y *JobPhase* también son datos del dispositivo, estos informan sobre el trabajo actual en el *Dispositivo* o la identidad del operador, si se ha registrado.

En la figura 7.10 muestra el código abreviado de la señal. *JobPhase* dice que el trabajo son 1002 hojas correctas y 71 desechadas. *Status = "Completed"* nos dice que ha finalizado.

El elemento *Detalles de MIS (MISDetails)* está como *Imponible (Chargeable)* y nos advierte que el cliente debe abonar el pedido (*CostType*), antes de retirarlo.

```

<JMF SenderID="_4004"...>
  <Signal ID="S1" Type="Status">
    <StatusQuParams JobID="_0001" JobPartID="_002"... />
    <DeviceInfo DeviceStatus="Idle" TotalProductionCounter="13862.0">
      <Device DeviceID="_4004" Status="Available"... />
      <Employee ID="_111" PersonalID="_013" Roles="Operator"
        Status="Available">
        <Person FamilyName="Cool" FirstName="Carl"... />
      </Employee>
      <JobPhase JobID="_001" JobPartID="_002"
        StartTime="2008-07-22T13:18:12+02:00" Status="Completed"
        TotalAmount="1002.0" Waste="71.0">
        <MISDetails CostType="Chargeable" WorkType="Original" />
        <Part SheetName="Blaetter" SignatureName="SIG1" />
      </JobPhase>
    </DeviceInfo>
  </Signal>
</JMF>

```

(*WorkType= Original*) indica que la orden se realizó sin cambios. La información del dispositivo (*DeviceInfo*), informa que el estado (*DeviceStatus*) es *Reposo (Idle)*.

Figura 7.10
Código de señal JMF

Fuera de Línea, Ajuste, Funcionamiento, Limpieza, Parado o Descocido son algunos estados que no mencionamos anteriormente. *StatusDetails* es un atributo más del mismo elemento que no aparece en la figura 7.10. Este elemento en particular nos informa del estado de las máquinas de imprimir offset. En la figura 7.11 enumeramos en una lista todas las tareas JDF y JMF.

Figura 7.11
Valores para el atributo
Detalles del Estado
(StatusDetails)
Cambio de Mantilla
Limpieza Mantilla
Avería
Limpieza de tintero
Control diferido
Tapa abierta
Limpieza del cilindro
Limpieza rodillos de mojado
Puerta abierta
Fallo
Formulario de Cambio
Componentes correctos
Limpieza rodillos de tinta
Mantenimiento
Área de salida llena
Limpieza de planchas
Reparar
Apagar
Cambiar tamaño
Cambio de manguito
Esperar aprobación
Calentamiento
Limpieza
Residuos

El último miembro de la familia JMF es el registro. La idea es la solicitud de comandos a terceras partes en determinadas circunstancias. Un ejemplo pueden ser los datos de preimpresión solicitados a un WMS por un MIS para enviarlos a una prensa en particular como información CIP3.

Esta jerarquía de mando necesita vías permanentes de comunicación (*PersistentChannel*) configuradas correctamente. Este tema es imprescindible para las señales y debemos ilustrarlo detalladamente como haremos en la siguiente sección.

Como ya habrá observado JMF se parece mucho a JDF y ambos son ampliables insertando contenidos propios como elementos, atributos y espacio de nombres. Posiblemente aumenta la flexibilidad de nuestro flujo de trabajo pero debemos contar con que puede reducir su compatibilidad.

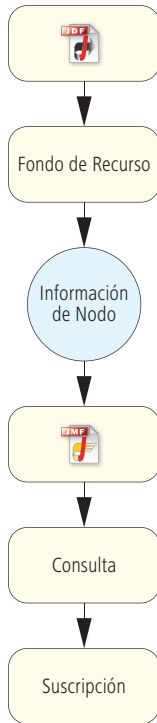


Figura 7.12
Suscripción solicitada con
la ayuda de Información de
nodo (*NodeInfo*)

7.3 JMF ICS

Las *Especificaciones de Conformidad de Interoperabilidad (ICS)* para JMF [12] establecen que las condiciones para los distintos módulos JDF/JMF se dividan en 2 niveles:

Nivel 1

- a) Datos JDF distribuidos mediante carpetas activas.
- b) Los gestores pueden establecer canales fijos de comunicación, para que los documentos JDF puedan incluir consultas y registros.

Nivel 2

- a) Los gestores pueden construir canales fijos de comunicación vía HTTP mediante mensajes JMF.
- b) Los datos JDF pueden distribuirse vía JMF.
- c) Existe comunicación bidireccional entre gestor y trabajador y pueden realizarse apuntes en las colas mediante JMF.

En la sección 6.5 presentamos el elemento *NodeInfo* con el ejemplo del gestor de programación para ejecutar un proceso. De hecho, pueden construirse canales de comunicación fijos con la ayuda de este recurso, como requiere el nivel 1b. Son necesarios para que el controlador se pueda suscribir a las señales JMF de los dispositivos que lo soporten. Un MIS también puede recibir información continua del estado y detalles de producción de una máquina de imprimir. La palabra mágica es el término suscripción y en el recurso *NodeInfo* se ha creado el elemento *Subscription*. En la figura 7.12 representamos la relación padre/hijo, donde vemos que la comunicación fija y válida se puede definir mediante una *Consulta (Query)*, siempre y cuando la suscripción sea aceptada o se hayan procesado los nodos JDF correspondientes.

Estos canales de comunicación fijos pueden enviarse como nodos JMF y tramitarse a través de HTTP, requisitos necesarios para cumplir el Nivel 2a. Los requisitos 2b y 2c ya se han presentado.

En la figura 7.7 hemos visto que las tareas JDF pueden transmitirse a los dispositivos utilizando el comando *SubmitQueueEntry* mediante HTTP.

Por último, queremos recordar que la complejidad de los envíos de mensajes en entornos JDF/JMF se incrementa de arriba abajo, con independencia de los niveles de partida de la ICS.

Estas son las diferentes posibilidades para los mensajes JMF:

- Sin mensajes JMF.
 - Señales JMF.
 - Consultas y respuestas JMF.
 - Comandos JMF.
 - Transmisión JDF vía HTTP.
-

Ejercicios.

Descargar e instalar un programa gratuito HTTP Sniffer de Internet. Analizar y leer paquetes HTTP en el Sniffer generados mediante el uso de un navegador.

8 Sistemas de Gestión de Órdenes

Los **Sistemas Gestores de Órdenes de Trabajo (OMS)** en la industria gráfica reciben varias denominaciones: planificación de recursos empresariales (ERP), sistema de gestión de información (MIS), sistemas de planificación y control de producción; también se usan soluciones específicas para industrias gráficas o simples hojas de cálculo y programas de contabilidad. Como es natural existen ciertas diferencias entre los términos, pero se utilizan indistintamente. A partir de ahora utilizaremos tanto MIS como OMS. Vamos a desglosar los puntos principales de estos programas para no quedarnos en los enunciados:

- Cálculo del coste final.
- Proceso desde el presupuesto hasta la facturación del pedido.
- Gestión de datos de clientes.
- Gestión de materiales.
- Integración con producción.
- Transferencia de detalles de producción.
- Programación de producción (gestión de recursos y entregas).
- Actividad de registro y recogida de datos.
- Seguimiento de las órdenes de trabajo.
- Integración con finanzas y contabilidad.
- Integración con clientes (portales empresariales e Internet).
- Integración con proveedores (proveedor de papel).

Normalmente, las imprentas que instalan un sistema de gestión de pedidos no adquieren todas las funciones disponibles, estas se suelen integrar opcionalmente, por ejemplo, un módulo para la planificación de producción. Algunos no son necesarios para el flujo JDF como la integración con finanzas y contabilidad de nóminas. Sin embargo los presupuestos, pedidos, clientes y materiales son esenciales para el sistema de gestión. En secciones posteriores analizaremos estos conceptos y la interdependencia existente entre ellos.

El interfaz del sistema gestor de producción es muy importante para el flujo JDF y lo estudiaremos en profundidad en la siguiente sección.

Este interfaz utilizado a la inversa, también puede recibir datos desde la planta de producción hasta el sistema gestor de pedidos, como el estado de producción o tiempo empleado en los trabajos. La Sección 8.4 trata la comunicación entre el comprador y el fabricante de productos impresos con PrintTalk.

8.1 Funciones básicas de un Sistema de Gestión de Pedidos

El tema principal de este libro es la información necesaria para realizar la oferta, no la creación de la oferta. En este punto el cálculo del precio real del producto es un misterio. Para estimar el valor del producto son muy importantes los datos de entrada y algunos de ellos pueden transmitirse a producción mediante JDF.

Para presupuestar un producto de impresión se necesita información de las siguientes áreas:

- Datos comerciales.
- Componentes del producto.
- Maquinaria de producción.
- Datos de material.
- Procesos de producción.

Al definir el producto, en multitud de casos los procesos y maquinaria necesarios para producción dependen de los datos de contenido que el impresor recibe del cliente, no se establecen únicamente en función de la preimpresión.

Los programas de presupuestos distinguen claramente el usuario de la administración. El Servicio de Atención al Cliente realiza el presupuesto, pero los datos esenciales están a cargo del administrador del sistema. En particular los datos maestros son:

- Mantenimiento del centro de costes.
- Inserción de costes y tiempos individuales (coste por puesto).
- Definición de los tipos de productos.
- Introducción de parámetros para la maquinaria de imprenta.
- Configuración de las bases de datos de clientes y materiales.

- Gestión de usuarios.
- Procesado de las formas de impresión y formularios.

Si los datos principales se configuran correctamente el usuario puede reunir rápidamente la información necesaria y evaluar los datos definidos previamente de manera sencilla. A continuación selecciona el cliente de la base de datos e inserta la persona de contacto, las direcciones de entrega y facturación o la información de solvencia y todo ello en el mismo trabajo. Luego se selecciona el tipo de producto (libro, folleto, tarjeta de visita, etc), la encuadernación, el tamaño de página y producto final o el color, sangrado y tirada. Los valores para productos más complejos se obtienen por separado. En un libro encuadernado, por ejemplo, el color de la cubierta puede ser diferente al del contenido. El usuario puede seleccionar en la pestaña de la sección "maquinaria de producción" los ajustes específicos para una prensa. Estos pueden tener predefinidos el tamaño de plancha, lado de la pinza, borde del papel, tira/retira, tamaño mínimo y máximo con respecto a la anchura del rodillo y las medidas de corte aunque lo habitual es tener un catálogo de plantillas. Al elegir el material de impresión se puede cargar el precio exacto desde la base de datos correspondiente, aunque el usuario puede anularlo o cambiarlo en cualquier momento.

Obviamente, el operador puede elegir la maquinaria que se utilizará para la producción pero también puede asignarla el sistema automáticamente. Por último, habrán ciertos pasos que el administrador del sistema no puede concretar al inicio y deben ser definidos posteriormente, pueden ser pruebas de color solicitadas por el cliente, equipos necesarios para procesar pasos inesperados o gastos de producción para obtener algún dato extra de impresión.

El sistema de gestión de pedidos utiliza esta información para calcular los costes de producción y generar un precio de venta. A continuación la oferta se pueden enviar por fax, correo electrónico o postal. Después de abrir el pedido, el cliente recibe una confirmación y se genera el Parte de Trabajo (JobTicket) necesario para el flujo de trabajo JDF.

El cálculo del coste final se utiliza para supervisar la realización del trabajo e identificar discrepancias con la orden. Con este cálculo se mejora la eficacia en producción y el ajuste del presupuesto. El término "coste final" no es totalmente cierto en este momento porque no se conoce el tiempo real de ejecución de los dispositivos y los materiales usados finalmente para cumplir el pedido.

Las tarifas horarias y costes de las distintas operaciones raramente se incluyen en el coste final o la base utilizada para la estimación. La recopilación de datos de planta si se incluye en el coste final. Estos se integran en el programa de contabilidad mediante los partes completados manualmente por los empleados y transferidos posteriormente. A menudo, los operarios introducen estos datos en los terminales de recopilación del taller en lugar de rellenar los partes de trabajo. Durante mucho tiempo se han usado para este fin módulos opcionales de los sistemas de gestión. Sin embargo, el objetivo real de un entorno JDF/JMF es la generación automática de datos de producción. Con JMF los dispositivos puedan enviar sus datos mediante protocolos estandarizados al centro de evaluación. Con los datos generados dinámicamente el análisis de costes es más exacto y resulta más barato y simultáneamente se pueden detectar fallos de dispositivo y consultar el estado de producción de los diferentes puestos.

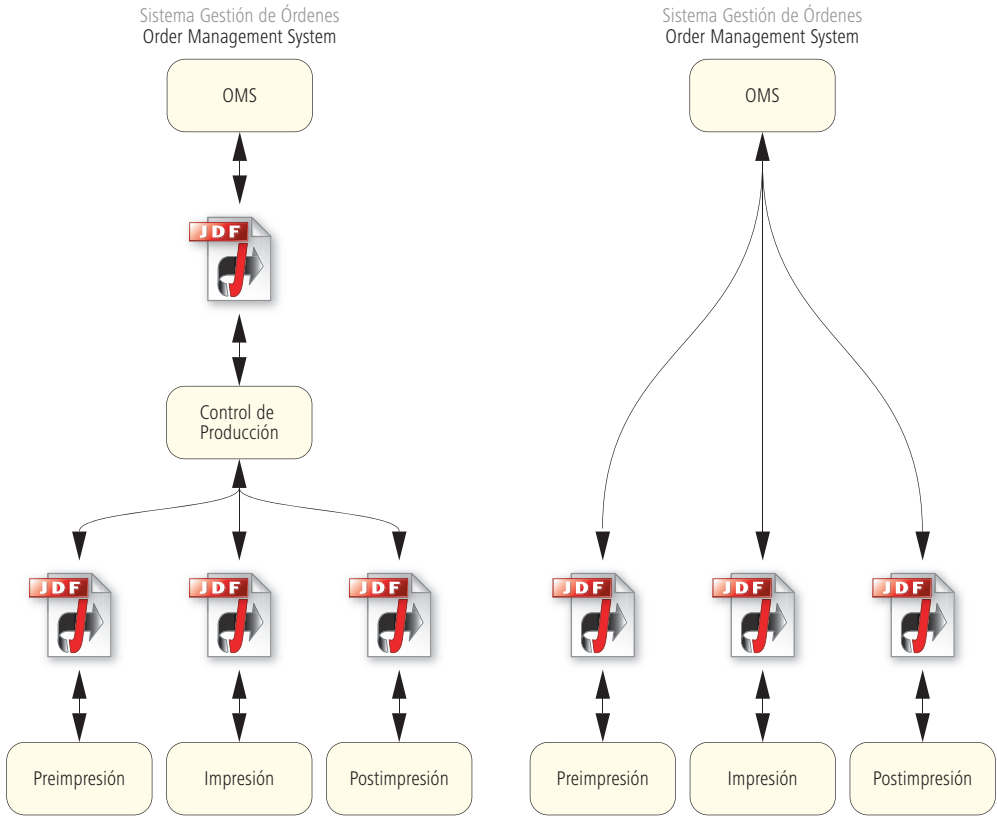
Pero en el mundo real JDF, los datos no siempre se obtienen automáticamente. Lo habitual junto a las áreas de trabajo manual y demás máquinas sin conexión es disponer de programas adecuados para el registro de tiempos, consumos y recursos generados. Estos datos se envían al MIS mediante JDF o JMF para su análisis posterior o ser utilizados en cualquier otra tarea.

8.2 Entornos de Producción

El interfaz más importante de todos los entornos JDF es el que se instala entre el MIS y producción. Es el más utilizado y también el más abierto por la gran cantidad de proveedores que deben comunicar a través de JDF/JMF sus sistemas de producción y gestión de pedidos.

En el capítulo 9 comprobaremos que no todos los interfaces JDF cumplen esta afirmación. En las secciones 2.2 y 2.3, hemos esquemático la comunicación entre JDF y preimpresión para los proveedores de servicios de impresión Y y Z.

Existen dos modos de intercambiar los datos JDF con producción. El sistema gestor de pedidos transmite directamente a los diferentes dispositivos en el primero. En el segundo, el MIS transfiere los datos al controlador central de producción JDF y este distribuye la información a los controladores y dispositivos de los demás departamentos, preimpresión, impresión y postimpresión (figura 8.1). Esta diferencia pequeña en apariencia tienen grandes implicaciones.



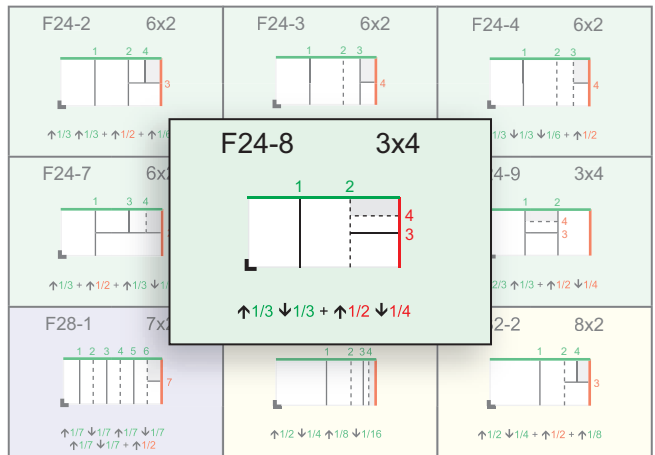
En la configuración de la izquierda producción tiene el control, mientras que en la derecha quien dirige es el propio MIS.

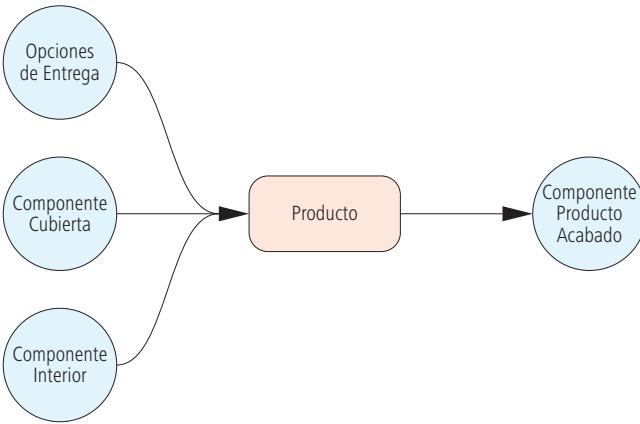
Figura 8.1 JDF entre MIS y producción

Por esta razón, el MIS debe ser capaz de recibir detalles de producción para reenviarlos al controlador o dispositivos de las plantas de impresión o postprocesado, como son valores preestablecidos de entintado por zonas, marcas de corte o secuencia de plegado.

Figura 8.2 Sección F24-8 de plegado de un catálogo en JDF

Por lo tanto, la exigencia del MIS va más allá de su tradicional función comercial. Los fabricantes ya han implementado estos 2 tipos de configuración (figura 8.1).





En el ejemplo de esta sección, presuponemos que existe un controlador central de producción JDF. Hasta ahora no hemos tratado la transmisión de datos JDF para máquinas de acabado, por lo tanto, dedicaremos el capítulo 11 a este interfaz.

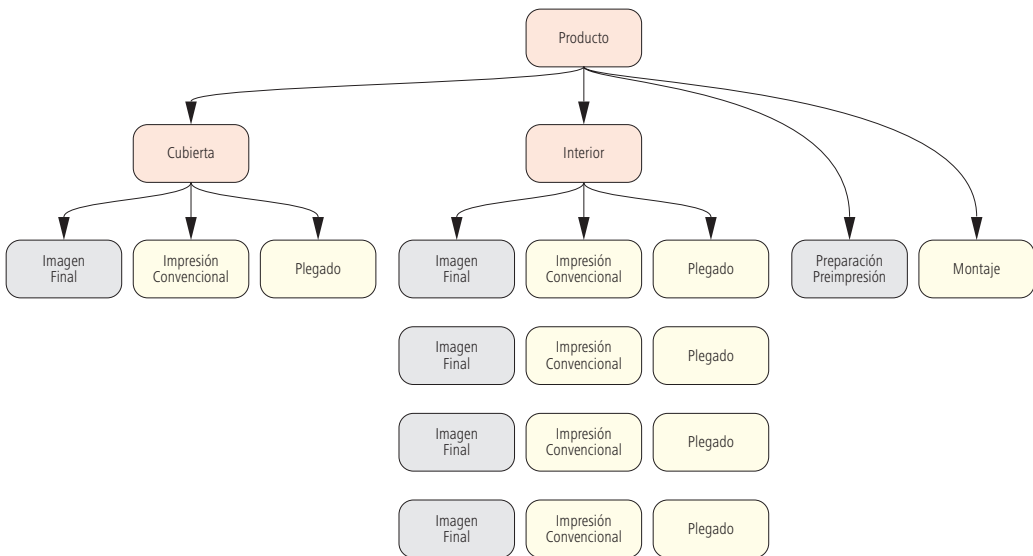
El interfaz es un elemento que se explica mejor con un ejemplo. El folleto encuadernado tiene 96 páginas en CMYK con un tamaño final de 12x12 cm.

Figura 8.3 Elemento JDF raíz de un folleto

Cubierta y páginas interiores se imprimen en hojas de 43x61 cm. Las firmas del interior se maquetan con la matriz de imposición descrita en F24-8 (tipo normalizado del catálogo de plegado) con 3x4 firmas (4x24=96 páginas). Las cubiertas se imprimen adoble cara, 6 por pliego (figura 8.2).

Figura 8.4 Estructura en árbol de un nodo JDF

En la figura 8.3, mostramos los componentes generados por el sistema de gestión de pedidos del archivo JDF. El nodo producto tiene 2 recursos de entrada tipo *Componente* denominados cubierta e interior. El producto final es un recurso de salida del mismo tipo *Componente*.



En la figura 8.4 vemos la estructura de árbol obtenida desde la perspectiva de las *Cajas Grises* y los nodos de producto y proceso. Los rectángulos rojos son nodos de producto, los amarillos nodos de proceso y los grises son *Cajas Grises* y por tanto nodos proceso de grupo. Las *Cajas Grises* de la categoría *Imagen Final* y de los procesos *Impresión Convencional* y *Plegado* se han repetido 4 veces porque se necesitan 4 firmas diferentes para el interior.

Algunos de los procesos y procesos de grupo ya fueron mencionados en la introducción a JDF pero antes de analizarlos a fondo describiremos sus características brevemente:

- **Preimpresión/Preparación (PrepressPreparation):**
Cajas Grises con todos los pasos necesarios para procesar los datos de contenido del trabajo mediante la imposición.
- **Montaje (Stripping).** Hoja de montaje.
- **Imagen Final (FinalImaging)** Esta *Caja Gris* incluye: *Imposición*, *Caja Gris* de *Rasterizado*, *Generación de Imagen Previa* y *Ajustes de Imagen*.

- **Impresión Convencional (ConventionalPrinting):**
Impresión con la máquina de imprimir (en este caso offset).

• **Plegado (Folding).**

El nodo proceso de grupo *Preimpresión/Preparación* tiene una estructura muy simple como vemos en la figura 8.5.



Figura 8.5
Recurso Caja Gris
preimpresión/preparación

Al recibir la orden, solo se sabe el número de páginas que suministra el cliente definido en el recurso de entrada *RunList*, la ubicación de los datos de contenido y el nombre se desconocen (figura 8.6).

```

    <RunList Class="Parameter" ID="_001"
    NPage="106" Status="Available" />
    
```

Figura 8.6
Recurso de entrada RunList

Pero el recurso de salida *Lista de Tarea (RunList)* contiene las páginas normalizadas con la transformación del espacio de color y sobreimpresión resueltas y preparadas para producción.

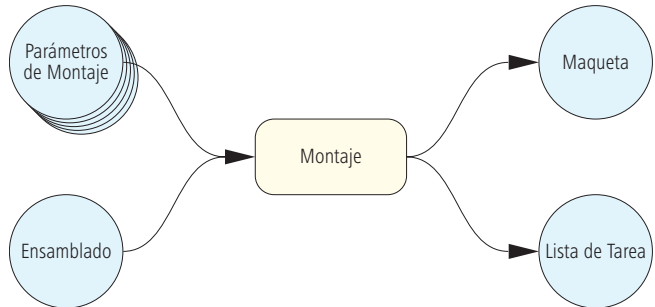


Figura 8.7
Proceso de imposición JDF

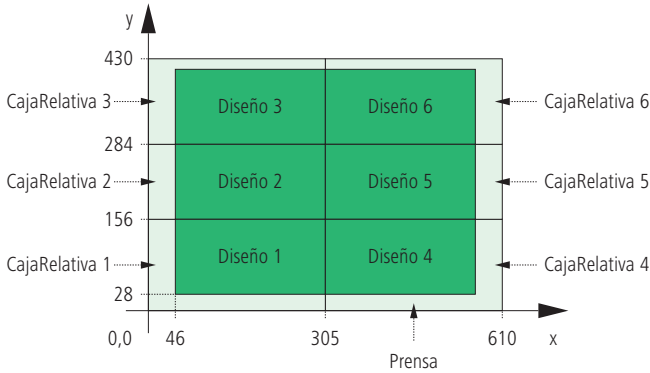


Figura 8.8 (arriba)
Parámetros de Montaje con
el plegado

Figura 8.9 (abajo)
Parámetros de Montaje con la
información de plegado (imposición)

En la figura 8.7 podemos ver el proceso de *Montaje (Stripping)* con todos sus recursos de entrada y salida.

Por cada signatura hay un recurso *Parámetros de Montaje (StrippingParams)* en total son 5 y contienen las posiciones exactas, en esencia es el plegado de las hojas en el pliego y un enlace a la plantilla de impresión (ver Figura 8.8).

```
<StrippingParams Class="Parameter" ID="_200" PartIDKeys="SignatureName
SheetName" PartUsage="Explicit" Status="Available">
  <StrippingParams SignatureName="SIG001">
    <StrippingParams SheetName="Cover" WorkStyle="WorkAndBack">
      <BinderySignatureRef rRef="_201" />
      <Position MarginBottom="79.370" MarginLeft="130.39"
MarginRight="0.0" MarginTop="0.0" Orientation="Rotate0"
RelativeBox="0.0 0.0 0.5 0.362" />
      <Position MarginBottom="0.0" MarginLeft="130.39" MarginRight="0.0"
MarginTop="0.0" Orientation="Rotate0"
RelativeBox="0.0 0.362 0.5 0.660" />
      <Position MarginBottom="0.0" MarginLeft="130.393" MarginRight="0.0"
MarginTop="51.023" Orientation="Rotate0"
RelativeBox="0.0 0.660 0.5 1.0" />
      <Position MarginBottom="79.37" MarginLeft="0.0"
MarginRight="130.393" MarginTop="0.0" Orientation="Rotate0"
RelativeBox="0.5 0.0 1.0 0.3627" />
      <Position MarginBottom="0.0" MarginLeft="0.0"
MarginRight="130.393" MarginTop="0.0"
Orientation="Rotate0" RelativeBox="0.5 0.362 1.0 0.660" />
      <Position MarginBottom="0.0" MarginLeft="0.0" MarginRight="130.393"
MarginTop="51.023" Orientation="Rotate0"
RelativeBox="0.5 0.660 1.0 1.0" />
      <StripCellParams BleedFace="7.086" BleedFoot="7.086"
BleedHead="7.086" BleedSpine="0.0" Spine="0.0" TrimFace="11.338"
TrimFoot="11.338" TrimHead="11.338" TrimSize="340.157 340.157" />
      <MediaRef rRef="_202">
        <Part SheetName="Cover" SignatureName="SIG001" />
      </MediaRef>
      <MediaRef rRef="_203">
        <Part SheetName="Cover" SignatureName="SIG001" />
      </MediaRef>
      <DeviceRef rRef="_204" />
    </StrippingParams>
  </StrippingParams>
  ...
</StrippingParams>
```

En la figura 8.9 hay una lista con la posición concreta de las 6 sig-naturas sobre el pliego, sin embargo, la información puede resul-tar algo confusa y describimos los detalles pormenorizados a con-tinuación.

El pliego se divide en 6 atributos cuyas dimensiones se encuentran en *Caja Relativa (RelativeBox)*. Los lados se describen con 4 valores entre 0.0 y 1.0, los 2 primeros son valores (x,y) a contar desde el margen inferior izquierda del rectángulo y los 2 siguientes son el margen superior y el derecho. El bloque inferior izquierda se define en el atributo *RelativeBox* = "0.0 0.0 0.5 0.362", la signatura infe-rior derecha también tiene valor 0, comienza en mitad del pliego en dirección X y termina a un 36,279% del pliego de impresión en dirección Y. El pliego tiene 430 mm de altura, por lo tanto la pri-mera *RelativeBox* tiene 156 mm en dirección Y (0.36279x430mm) y corresponde con la hoja de plegado 1 (ver figura 8.8). Los de-más se calculan igual.

Como podemos ver en la figura 8.10 cada *RelativeBox* alberga sus datos respectivos, los valores son los márgenes *Inferior, Izquierdo, Derecho y Superior*. Un buen ejemplo de recurso dividido puede ser el recurso *StrippingParams*, posiblemente sea más válido que el de la figura 6.14. Sus componentes son: *Clave ID Parcial (PartIDKeys)*, *Signatura (SignatureName)* y *Hoja (SheetName)*.

También debemos explicar el código JDF de los demás compo-nentes de la figura 8.9. En el elemento *Parámetros de Celda (StripCellParams)* se definen los datos del primer corte y el ta-maño del producto parcial, ambas referencias a recursos *Medio* son respectivamente el sustrato y la plancha. El recurso *Signatura de Encuadernación (BinderySignature)* se define en la cuarta fila como vemos en la figura 8.11 Obviamente, es la plantilla de impo-sición y una entrada del *Catálogo de Plegado (FoldCatalog)*.

Como alternativa se puede introducir la plantilla de imposición di-rectamente en el subelemento *SignatureCell* de *BinderySignature*, este recurso corresponde a una plantilla de plegado estandarizada aplicable a múltiples hojas de plegado; se define independientemente del tamaño de las páginas a partir de la parte supe-rior del pliego.

Figura 8.10
Posición de los rectángulos
en la Caja Relativa
(RelativeBox)

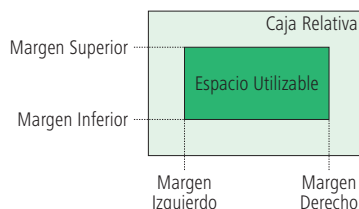


Figura 8.11
Código del Recurso
Diseño de Encuadernación
(BinderySignature) con la
plantilla de imposición

```
<BinderySignature Class="Parameter" DescriptiveName="F04-01_ui_2x1"
  FoldCatalog="F4-1" ID="_201" NumberUp="2 1" Status="Available" />
```

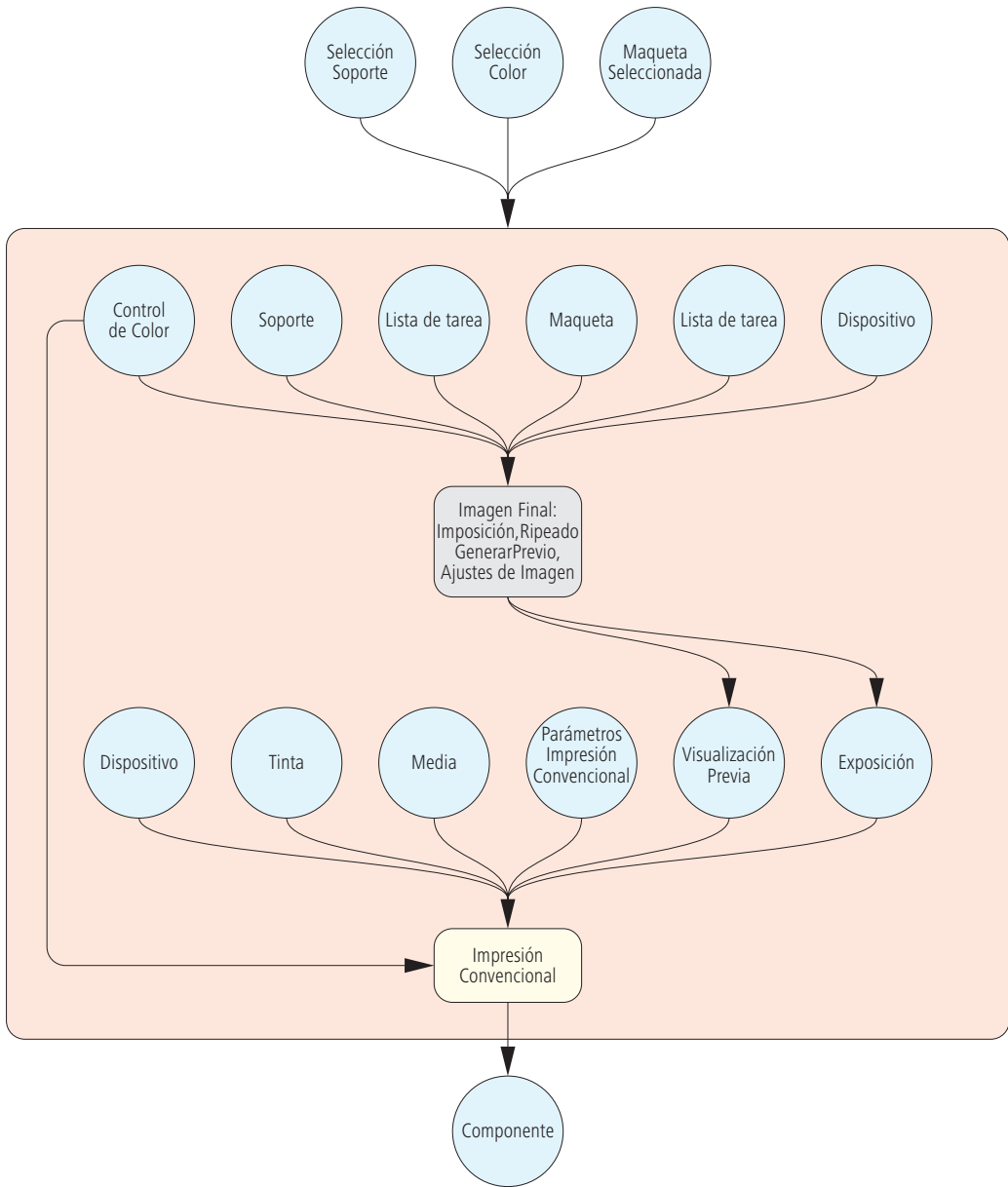


Figura 8.12
Nodo de producto
"Cubierta" con su
infraestructura y recursos

Sin embargo, el recurso *Ensamblado (Assembly)* solo se menciona una vez en el documento JDF. Este recurso, introducido en la versión JDF 1.2, recoge toda la información de alzado y embuchado de las hojas.

En la figura 8.12 esquematizamos el nodo de producto representado por la *Caja Gris "Cubierta"* con el proceso y sus vínculos.

Los 4 recursos fuera del cuadro rojo proporcionan la entrada y recursos de salida del nodo preferencias de producto. Los 3 situados en la parte superior sirven de entrada y describen el producto final previsto: *LayoutIntent* es el tamaño final, *ColorIntent* el color y *MediaIntent* el sustrato. El recurso de salida inferior *Componente* representa el subproducto "Cubierta". Este es a su vez, la entrada de los nodos superiores que definen el producto final.

La *Caja Gris* y los procesos están unidos por un grupo de recursos. Estos no necesitan estar en los nodos del producto para describir la cubierta, pueden estar en el elemento raíz del nivel superior o del producto global. Las flechas entre los recursos y *Cajas Grises* solo son *Enlaces de Recursos (ResourceLinks)*.

El diagrama de la figura 8.12 puede parecer erróneo a simple vista porque el recurso de entrada de la *Caja Gris Lista de Tarea (RunList)* aparece dos veces pero es intencionado porque el contenido de los recursos es diferente de los propios recursos.

Esta *Caja Gris* en concreto, contiene los siguientes recursos de entrada:

- **Dispositivo (Device):** Definición de la filmadora de planchas.
- **Lista de Tarea (RunList):** Archivo con los datos de contenido procesados, por tanto la salida de Preparación/Preimpresión.
- **Maqueta (Layout):** Signaturas del producto final.
- **Lista de Tarea (RunList):** Archivo con las marcas de la hoja.
- **Soporte (Media):** Definición de las planchas.
- **Control de Color (ColorControl):** Colores de las signaturas.

Algunos recursos contienen características del producto íntegro, no solo del Componente "Cubierta". El elemento JDF del producto final debe contener los recursos *Layout*, *RunList* y *ColorantControl* para que el subproducto "Interior" pueda referirse a ellos. De los recursos mencionados, realmente solo deben estar *Disponibles: Dispositivo* y *ColorControl*. Los demás están en uso pero sin entradas importantes, por lo tanto, su estado es *No disponible*.

El sistema gestor de pedidos, por ejemplo, desconoce el nombre del archivo y la ubicación de los datos de contenido y, además, la selección de la plancha no se producirá hasta que esté en producción.

La Caja Gris “*Imposition RIPing PreviewGeneration ImageSetting*” no se ha descrito completamente. Se han tratado los recursos de salida necesarios para el siguiente proceso como entradas, estos son *ExposedMedia* y *Preview*. Su estado debe ser *Disponibile* porque no se ha ejecutado la *Caja Gris*, para ser más concretos diremos que la *Caja Gris* no está en marcha y las tareas no se pueden ejecutar. Como mencionamos en el capítulo 6, en principio, las *Cajas Grises* no son ejecutables.

Impresión Convencional dispone de los siguientes procesos como recursos de entrada:

- **Soporte Expuesto (ExposedMedia):** Plancha para el trabajo.
- **Previsualización (Preview):** Imágenes previas por separación (para preajuste de zonas de entintado).
- **Parámetros Impresión Convencional (ConventionalPrintingParams):** Tirada y tipo de imprenta (bobina, hoja).

Figura 8.13
Elementos estructurales
del producto parcial
“cubierta”

```
<JDF DescriptiveName="Cover" Status="Waiting" Type="Product"...>
  <ResourceLinkPool>
    ...
  </ResourceLinkPool>
  <ResourcePool>
    ...
  </ResourcePool>

  <JDF Category="FinalImaging" DescriptiveName="Cover (CTP) "
    Status="Waiting" Type="ProcessGroup"
    Types="Imposition RIPing PreviewGeneration ImageSetting"...>
    <ResourceLinkPool>
      ...
    </ResourceLinkPool>
    <ResourcePool>
      ...
    </ResourcePool>
  </JDF>

  <JDF DescriptiveName="Cover" Status="Waiting"
    Type="ConventionalPrinting"...>
    <ResourceLinkPool>
      ...
    </ResourceLinkPool>
    <ResourcePool>
      ...
    </ResourcePool>
  </JDF>
</JDF>
```

```
<ResourceLinkPool>
  <ComponentLink Usage="Output" rRef="_100" />
  <LayoutIntentLink Usage="Input" rRef="_101" />
  <ColorIntentLink Usage="Input" rRef="_102" />
  <MediaIntentLink Usage="Input" rRef="_103" />
</ResourceLinkPool>
```

- **Soporte (Media):** Información sobre el sustrato o soporte.
- **Tinta (Ink):** Información sobre los colores.
- **Dispositivo (Device):** Información de la máquina de imprimir.
- **Control de Color (ColorantControl):** Colores individuales.

Figura 8.14
Fondo Enlace de Recurso
(ResourceLinkPool) del
nodo JDF "cubierta"

En este caso, todos los recursos tienen estado *No disponible*, incluso *Tinta*, *Dispositivos* y *Control de Color*.

En la figura 8.13, mostramos la estructura de nodos JDF del subproducto "cubierta" y solo contiene recursos y enlaces a otros recursos. Lo que hemos visto en el diagrama de la figura 8.12 no importa, lo que está claro es que el *Fondo de Enlace de Recursos* (ResourceLinkPool) del nodo "Cubierta" debe estar visible como se muestra en la figura 8.14. Los otros ResourceLinkPool se construyen del mismo modo. También debemos mencionar los propios recursos. En la figura 8.15 vemos el recurso *Tinta (Ink)* del anverso (*Ink Side= Front*), los colores del reverso de cada signatura están por definir. Se describen a sí mismos como recursos divididos mediante el valor de los atributos *PartIDKeys*.

Figura 8.15
Separaciones requeridas
para el producto parcial

```
<Ink Class="Consumable" ID="_104" PartIDKeys="SignatureName SheetName Side
Separation" PartUsage="Implicit" Status="Available">
  <Ink SignatureName="SIG001">
    <Ink SheetName="Cover">
      <Ink Side="Front">
        <Ink Separation="Cyan" />
        <Ink Separation="Magenta" />
        <Ink Separation="Yellow" />
        <Ink Separation="Black" />
      </Ink>
      <Ink Side="Back">
        <Ink Separation="Black" />
      </Ink>
    </Ink>
  </Ink>
  Y así sucesivamente para las otras 4 signaturas del interior
</Ink>
```

Name	Data Type	Description
<i>BillingCode</i> ?	string	A code to bill charges incurred while executing the Node.
<i>CustomerID</i> ?	string	Customer identification used by the application that created the Job. This is usually the internal customer number of the MIS system that created the Job.
<i>CustomerJobName</i> ?	string	The name that the customer uses to refer to the Job.
<i>CustomerOrderID</i> ?	string	The internal order number in the system of the customer. This number is usually provided when the order is placed and then referenced on the order confirmation or the bill.
<i>CustomerProjectID</i> ? New in JDF 1.2	string	The internal project id in the system of the customer. This number might be provided when the order is placed and then referenced on the order confirmation or the bill.
<i>rRefs</i> ? Deprecated in JDF 1.2	IDREFS	Array of <i>IDs</i> of any Elements that are specified as <i>ResourceRef</i> Elements. In version 1.1 it was the <i>IDREF</i> of a <i>ContactRef</i> . In JDF 1.2 and beyond, it is up to the implementation to maintain references.
Company ? Deprecated in JDF 1.1	refelement	Resource Element describing the business or organization of the contact. In JDF 1.1 and beyond, Company affiliation of Contacts is specified in Contact .
Contact * New in JDF 1.1	refelement	Resource Element describing contacts associated with the customer. There SHOULD be one Contact [contains (@ <i>ContactTypes</i> , " <i>Customer</i> ")]. Such a Contact specifies the primary customer's name, address etc.
<i>CustomerMessage</i> * New in JDF 1.2		

Figura 8.16
Requisitos del recurso
CustomerInfo en la
especificación JDF

Evidentemente no aparecen todos los recursos de "cubierta", los analizaremos más adelante junto al nodo JDF "Interior".

El recurso *CustomerInfo* contiene datos del cliente y otros recursos adicionales que se han ocultado deliberadamente por su parecido con los casos mostrados en las figuras 5.2 y 5.3. Ahora dejaremos momentáneamente el código y sus ejemplos. En su lugar definiremos el recurso *CustomerInfo* según la especificación 1.4, para poder comprender las especificaciones ICS con más facilidad.

Las tablas descriptivas de las especificaciones se construyen básicamente como muestra la figura 8.16; en la columna de la izquierda se sitúan los elementos (atributos, subelementos o referencias a recursos), en la columna central está el tipo de datos y su descripción a la derecha.

En la primera columna resaltan los comentarios en azul, estos son "New in JDF 1.2" o "Deprecated in JDF 1.1", este último significa que el elemento estructural se ha desechado desde la versión 1.1, así que no aparece en el código de versiones JDF superiores. Obviamente aunque no debemos utilizar "conceptos obsoletos", tienen que permanecer en las especificaciones por compatibilidad. Además, vemos algunos caracteres especiales en los nombres, como el signo de interrogación (?) o el asterisco (*).

En otras partes de la especificación, encontramos el signo más (+) (ver ejemplo de la figura 12.8) o algún otro carácter, que nos indican el número de instancias posibles:

? Opcional.

* Varias veces o nunca.

+ Varias veces o una.

Si no hay caracteres especiales en el nombre, el elemento estructural aparece una sola vez, por lo tanto, el subelemento *Contact* puede aparecer una, varias veces o nunca pero el atributo *CustomerID* solo puede aparecer una vez.

El Tipo de Datos *String* es una secuencia de caracteres. *RefElement* hace referencia a un elemento o es un elemento. Cuando aparece *Element* debe estar accesible (ausente en la figura 8.16).

A continuación explicamos los elementos vigentes en la Versión 1.4:

- **BillingCode**: número de cuenta para el trabajo.
- **CustomerID**: número de cliente del MIS.
- **CustomerJobName**: nombre del Cliente.
- **CustomerOrderID**: número de orden del Cliente.
- **CustomerProjectID**: número de proyecto del Cliente.
- **Contacto**: Referencia al Elemento que describe un contacto.
- **CustomerMessage**: un mensaje al cliente. Por ejemplo, un correo electrónico automático al cliente al generarse el nodo.

Los elementos *Contact* y *CustomerMessage* tienen su tabla en la especificación. Ambos tienen subelementos especificados en otras tablas (*Address*, *ComChannel*, *Company* y *Person*), tiene hipervínculos que no mostramos para no confundir al lector. Algunos elementos pueden ser empleados en elementos padre diferentes pero solo pueden aparecer una vez en la especificación. Por ejemplo, *ComChannel* es simultáneamente subelemento de *Contact* y de *CustomerMessage*. En realidad, el lector debe seguir los enlaces inferiores (los elementos hijo posibles) y los enlaces superiores, donde se encuentran los denominados "recursos abstractos" que describen los atributos y todos los recursos que puedan tener. Estos atributos generales (por ejemplo *ID*, *Class* o *Author*) no aparecen en la tabla de ningún recurso en concreto. Esta función es aplicable a todos los nodos JDF.

Hasta ahora, tenemos cubierto el flujo de datos prácticamente completo desde el MIS hasta preimpresión, pero no lo hemos hecho en sentido contrario y para el cálculo del coste final resulta crucial. Para esta comunicación disponemos de dos posibilidades que en muchas ocasiones fluyen en paralelo:

- El MIS recibe mensajes JMF desde el sistema de producción.
- El JDF modificado en el sistema de producción se devuelve al MIS.

Como ya explicamos en el capítulo 6 cuando los JDF son devueltos por los sistemas de producción al MIS este consulta los informes contenidos en las Auditorías (*AuditPools*).

8.3 Documentos ICS MIS

La importancia del interfaz entre el sistema de gestión/MIS y producción queda demostrada por el gran número de especificaciones ICS publicadas (figura 6.7). La versión vigente en la fecha de redacción de este libro era la 1.3, pero la vigencia de las especificaciones es muy corta y el lector debería consultar en internet la versión más reciente para su descarga [12].

- *MIS ICS*
- *MIS a Preimpresión ICS*
(*MIS to Prepress ICS*)
- *MIS a Impresión Convencional/Pliego ICS*
(*MIS to Conventional Printing/Sheetfed ICS*)
- *Prensa: MIS a Impresión en Bobina ICS*
(*Newspaper: MIS to WebPress ICS*)
- *MIS a Acabado ICS*
(*MIS to Finishing ICS*)

Explicaremos la primera especificación *MIS ICS* y presentaremos segmentos de las demás en los capítulos 9 a 11.

MIS ICS no trata documentos específicos de preimpresión, impresión y postprocesado, describe los requisitos generales del MIS y los diferentes niveles necesarios para su cumplimiento que, como podremos observar, resultan muy similares al *JMF ICS*. Existen 3 niveles que afectan directamente a los mensajes JMF; en esta sección solo presentaremos los dos primeros (figura 8.17).

- **Nivel 1:** El MIS entrega el JDF al *Trabajador* vía carpeta activa. (Figura 8.17 Nivel1).
- **Nivel 2:** El MIS puede generar consultas JMF mediante el elemento *NodeInfo* para crear un *Canal de Comunicación Constante* del *Trabajador* al MIS (*PersistentChannel*). (Figura 8.17, Nivel 2a). Así se retransmite la señal del *Trabajador* al MIS. Este método es nivel 1 *JMF ICS*. En este caso el *Trabajador*, no necesita ser servidor HTTP, ni procesar paquetes HTTP. En la figura 8.17 podemos ver la alternativa Nivel 2b, el MIS envía directamente los mensajes JMF. Aquí el *Trabajador* puede ser un servidor HTTP y cumplir el Nivel 2 del *JMF ICS*.

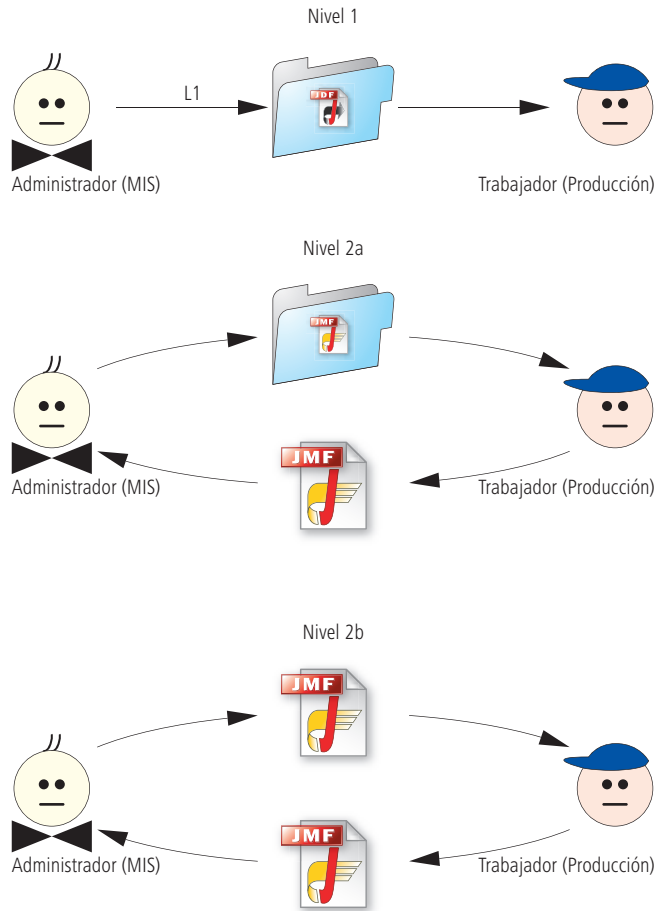


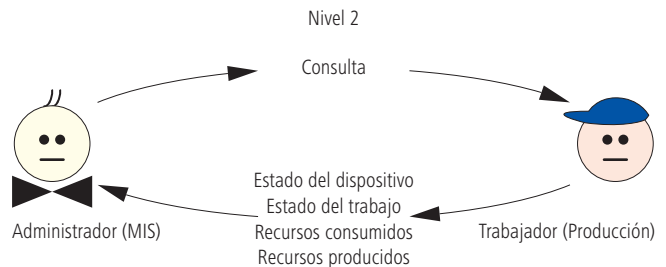
Figura 8.17 Nivel 1 y 2 del MIS ICS

Usar JMF resuelve otros requisitos como por ejemplo cuando un MIS *Consulta* al *Trabajador* sobre el estado del dispositivo o la tarea y los recursos consumidos/producidos hasta el momento y este le informa sobre todas la cuestiones planteadas (figura 8.18).

Figura 8.18 El Trabajador, al recibir la solicitud, informa al Administrador sobre trabajos y recursos.

El software de gestión/MIS que cumple con el Nivel 2 del *MIS ICS*, también debe estar capacitado para enviar las consultas correspondientes vía *NodeInfo* o directamente mediante mensajes JMF.

Un *Trabajador* Nivel 2 *MIS ICS* recibe, interpreta y responde a las *Consultas* sobre trabajos y dispositivos, por lo tanto cumplir con el Nivel 2 *MIS ICS* es un requisito del software MIS y del controlador/dispositivos para el seguimiento de los trabajos y estado de los dispositivos.



En la práctica hay implementaciones de flujos de trabajo JDF que no disponen de dichas funciones. Los requisitos de los elementos JDF no se definen solo en el *MIS ICS*, los propios JDF también deben cumplirlos y necesitamos concretar los detalles más pequeños, por ejemplo, es muy importante conocer el *Nivel* y la *Versión ICS*. El MIS incluye en cada documento JDF para este fin, un atributo especial en su elemento raíz. Este aparte, debe contener dos recursos de entrada especiales: *CustomerInfo* con los datos del cliente y *NodeInfo* con las fechas y mensajes JMF opcionales. Cada nodo JDF raíz (*JDFroot*) generado por el MIS es un nodo de producto y sus propiedades se almacenan en *Preferencias* del recurso (*Intent*) y cada uno debe tener su propio recurso de entrada *Intent*. Si carece de él es un producto sin propiedades ni utilidad. Ya hemos comentado que cada elemento raíz JDF debe tener relación con un componente del recurso de salida, representante del producto final.

Al finalizar la tarea cada *Trabajador* debe registrar el inicio, duración y la finalización de los procesos en los *nodos de proceso JDF* para mantener actualizado el elemento *Audit*.

En el software *MIS ICS Nivel 2* se añaden datos adicionales al elemento de *AuditPool Auditoría de Recursos (ResourceAudit)*; es la información sobre los materiales utilizados por el *Trabajador* durante la ejecución del proceso (planchas, papel, tinta, etc.).

8.4 PrintTalk/JDF Interfaz con el Cliente

La Organización PrintTalk, auspiciada por la Asociación de Proveedores de Impresión, Publicaciones y Tecnología de Transformación NPES, publicó en el año 2000 la especificación PrintTalk basándose en cXML, como ya comentamos en la sección 5.4. Esta organización independiente se disolvió en 2004 para integrarse en CIP4.

Los documentos JDF enviados por el cliente pueden ser importados a los sistemas de gestión de pedidos para generar nuevos presupuestos u órdenes.

El cliente puede crear documentos JDF en Acrobat describiendo el producto mediante nodos. La imprenta, a continuación, puede importarlo a su sistema. Por esta razón, los sistemas gestores de pedidos deben ser capaces de recibir la descripción del producto desde el exterior. El problema de este modelo es que no puede tratar las operaciones comerciales adecuadamente porque el nodo de identificación del trabajo (*JobID*) ha sido creado por el cliente y no coincide con la numeración interna de la imprenta.

Es mucho más lógico que la imprenta inicie la orden de trabajo en su software MIS y envíe el archivo JDF vacío al cliente por correo electrónico. El cliente completa el pedido con los datos en Acrobat y lo devuelve a la imprenta para asignarlo automáticamente a la orden existente.



Figura 8.19
Comunicación PrintTalk
entre cliente e impresor

Los clientes finales, delegaciones y proveedores de servicios gráficos pueden comunicar a la imprenta todos los parámetros del pedido mediante JDF. De este modo se pueden coordinar los datos automáticamente desde el sistema al cliente y viceversa (por ejemplo, las fecha de entrega y/o número de copias). Estos métodos de retroalimentación electrónica son muy rentables y PrintTalk mejora la comunicación JDF entre cliente e imprenta.

En la figura 8.19 ilustramos este principio. Las ofertas, solicitudes y objetos de negocio se intercambian entre cliente y proveedor a través de internet. También se pueden transmitir otros documentos, como pedidos, confirmaciones y facturas.

La integración del interfaz PrintTalk/MIS simplifica muchos trámites administrativos. Es muy parecido a un Portal Web o sistema Web to Print.

Para llevarlo a cabo se utilizan 2 configuraciones básicas:

El cliente comunica con el servidor Web del impresor directamente o existe un intermediario entre las partes. En este caso el cliente entra en el servidor Web del intermediario, este solicita precios a diferentes imprentas vía PrintTalk. Si las solicitudes se generan automáticamente provocan ofertas en tiempo real, si las gestionan personas tienen cierta demora. El intermediario recibe las ofertas como objetos PrintTalk, puede analizarlas, reedita lo que considera oportuno y, finalmente, reenvía las ofertas al cliente en línea. Este procedimiento es el que utilizan, por ejemplo, las agencias de viajes. Las cotizaciones automáticas pueden provocar guerras de precios.

Si la comunicación entre cliente y proveedor es directa, la técnica es distinta. En la primera posibilidad el cliente genera los documentos PrintTalk en su navegador y los envía al servidor del impresor. En la segunda, el servidor Web to Print del impresor gestiona la comunicación PrintTalk y comparten una conexión estándar por internet donde PrintTalk es el protocolo del interfaz *MIS/WebtoPrint* del impresor, como vemos en la figura 8.20.

La red local del impresor necesita un área protegida por un firewall (cortafuegos, filtro de entradas/salidas de sistema conectado a red). El servidor de la tienda electrónica es la "zona de distensión", el cliente potencial comunica con el sistema de pedidos en línea de la imprenta por internet, este genera los datos PrintTalk y los transmite al MIS. Por esta razón desde el exterior se solicitan productos predefinidos en HTML, pero el MIS recibe las descripciones de los productos en JDF y los elementos *BusinessObject* en PrintTalk.

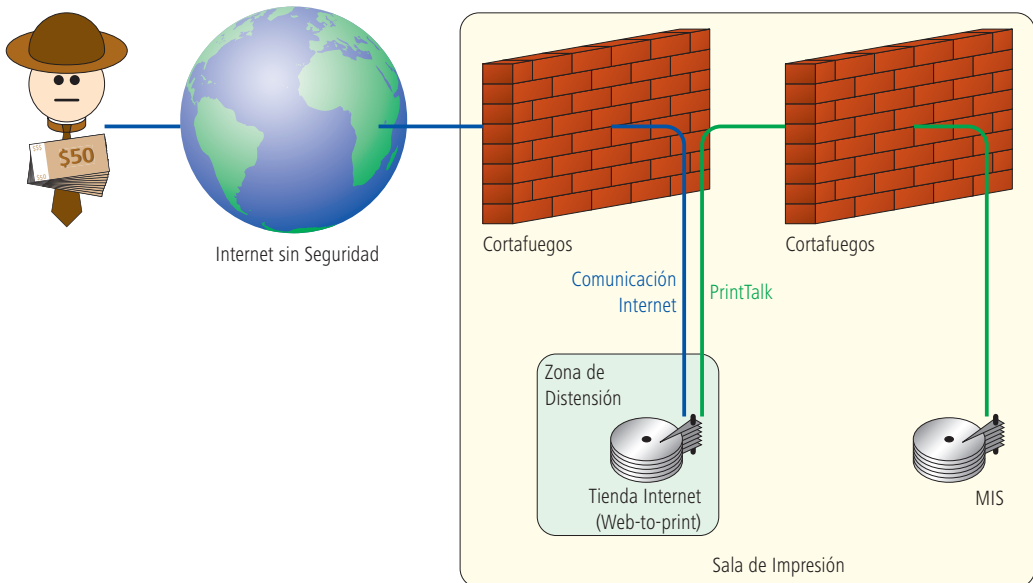
Esta configuración soporta PrintTalk y JDF entre el MIS y el portal Web con un interfaz de comunicación independiente.

PrintTalk permite establecer comunicación *Intermediario/Imprenta (Business-To-Business)* o *Imprenta/Cliente (Business-To-Customer)*. Lo cierto es que hay pocas aplicaciones PrintTalk, pero analizaremos la forma de implementar un sistema.

Posiblemente, los proveedores son reticentes a PrintTalk porque no les interesa un interfaz abierto para conectar con el MIS y prefieren ofrecer sus propias soluciones Web.

En la figura 8.21 mostramos un documento PrintTalk estructurado de manera muy similar a la estructura cXML de la figura 5.7. El encabezado PrintTalk se construye igual que en cXML. En PrintTalk, también existe un elemento solicitud de pedido. Los subelementos son distintos; solo tienen un *Objeto de Negocio (BusinessObject)*.

Figura 8.20
Servicio de impresión web
y MIS; el protocolo PrintTalk
se utiliza dentro de la
tienda de impresión



En el ejemplo mostramos una orden de compra pero los *Objetos de Negocio* posibles son:

- **Quotation:** Estimación, Presupuesto
- **PurchaseOrder:** Orden de compra.
- **Confirmation:** Confirmación del pedido.
- **Cancellation:** Cancelación del objeto de negocio.
- **Refusal:** Objeto de negocio denegado.
- **OrderStatusRequest:** Consulta de estado del pedido
- **OrderStatusResponse:** Respuesta a la consulta sobre el estado del pedido.
- **ProofApprovalRequest:** Solicitud de aprobación.
- **ProofApprovalResponse:** Prueba aprobada o rechazada.
- **Invoice:** Factura.
- **ReturnJob:** Devolución del trabajo al cliente.

Cada *BusinessObject* define la transacción comercial con sus propios atributos. Por ejemplo, al crear el pedido se seleccionan entre otras las subcategorías moneda, forma de pago o precio. La figura 8.22 muestra dos precios, uno es la producción y otro los portes. En la versión JDF anterior se permitía información de precio y forma de pago en recursos *DeliveryIntent*. Actualmente, se ha retirado toda esta información de JDF y se ha incorporado en los elementos *PrintTalk* correspondientes.

```
<PrintTalk...>
  <Header>
    <From>
      ...
    </From>

    <To>
      ...
    </To>

    <Sender>
      ...
    </Sender>
  </Header>

  <Request>
    <PurchaseOrder...>
      <jdf:JDF ...>
        <jdf...>
          ...
        </jdf>
        <jdf...>
          ...
        </jdf>
      </jdf:JDF>
    </PurchaseOrder>
  </Request>
</PrintTalk>
```

Figura 8.21
Estructura de un
documento *PrintTalk*

Figura 8.22
Orden con *PrintTalk*

```
<PurchaseOrder AgentID="CC" AgentDisplayName="CarlCool"
RequestDate="2008-11-13T11:00Z" BusinessID="A001" Currency="EUR"
Expires="2008-12-13T11:00Z" >
  <Pricing>
    <Price LineID="_1" DescriptiveName="Hard Cover Books" Amount="6800"
      Price="15000.00" />
    <Price LineID="_2" DescriptiveName="Shipping" Price="980.00"/>
  </Pricing>
  <jdf:JDF...>
    <JDF...>
      ...
    </JDF>
  </jdf:JDF>
</PurchaseOrder>
```

Algunos de estos *Objetos de Negocio (BusinessObjects)* contienen uno o varios nodos JDF. En la figura 8.22, el elemento *Orden de Compra (PurchaseOrder)* contiene tres nodos JDF, pero solo se representa el nodo superior.

La descripción del producto se encuentra en el elemento JDF. Los subproductos del pedido, *Información de Cliente (CustomerInfo)* o *Fecha de Entrega (DeliveryIntent)*, por ejemplo, se especifican en otros nodos JDF pero deberían estar contenidas en *ResourcePool* del elemento raíz JDF. Este ejemplo ya se ha descrito en las últimas secciones y en el capítulo 6 por lo tanto no nos extenderemos más. Además, se ha construido cumpliendo las estrictas normas de JDF.

Ejercicio:

Abrir Acrobat y generar un archivo JDF. Completar los datos del cliente, material y producto, este puede ser un folleto de 60 páginas con el interior y la cubierta en CMYK y añadir algún color plano a esta última.

Descargar JDF Editor de CIP4.org. Abrir el archivo JDF producido anteriormente y analizar la estructura.

Cambiar el nombre del archivo **.jdf* a **.xml* para verlo en un navegador. Examinar los datos JDF e identificar los valores introducidos en Acrobat.

9 Preimpresión

El mundo JDF/JMF se parece a la figura 9.1 desde la perspectiva de preimpresión: cada flecha es una conexión potencial JDF/JMF. El sistema WMS de preimpresión generalmente, recibe los datos de entrada JDF del sistema gestor de órdenes de trabajo MIS (flecha 1). Según la configuración instalada, preimpresión puede enviar mensajes de vuelta al sistema de gestión. Ya hemos visto que los datos JDF entregados por el MIS en ocasiones no son suficientes para realizar la producción (palabra clave: ¡Caja Gris!). Significa que los diferentes módulos (flecha 2) de preimpresión crearán nuevas estructuras JDF y mensajes JMF opcionales. Del sistema gestor de pedidos, se obtiene una sección del JDF y junto a los nuevos datos generados en preimpresión se transmite a impresión (flecha 3) y a postimpresión (flecha 4), directamente o mediante el MIS.

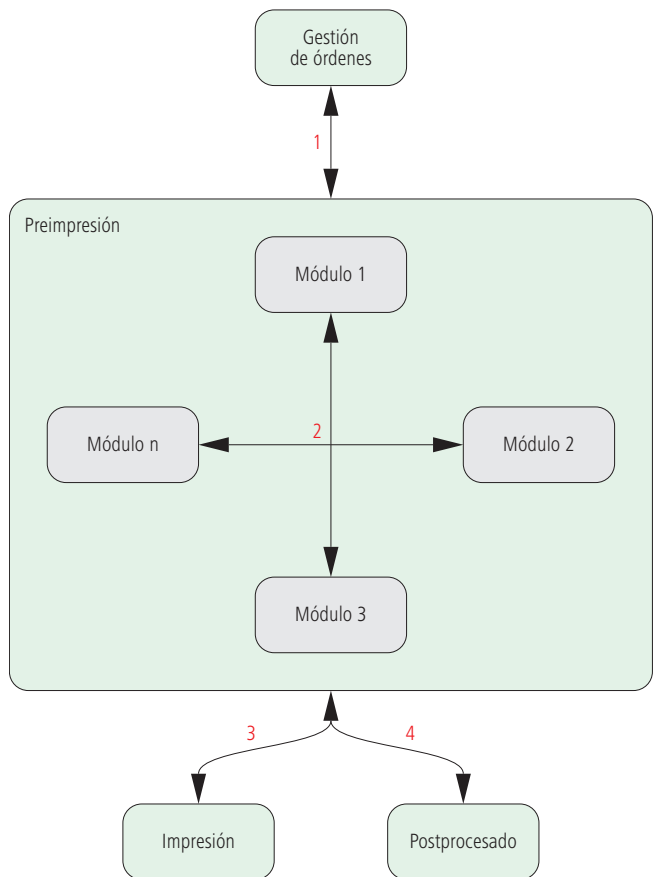
En este capítulo desarrollaremos los interfaces 1 y el 2. El 3 y el 4 serán analizados en los capítulos 10 y 11, correspondientes a impresión y postimpresión, respectivamente.

MIS a Preimpresión ICS define las características del interfaz 1, tratado en la sección 9.1. En las secciones 9.2 a 9.5 explicamos mediante ejemplos las características internas del interfaz 2.

Antes vamos a recordar los pasos de preimpresión revisados en la sección 3.2. Ya hemos dicho anteriormente que preimpresión recibe los datos en PDF, por lo tanto, diseño y creación no están en el listado. Estas tareas son habituales en la imprenta, pero su tratamiento es un poco distinto.

Se puede utilizar una prueba de pantalla o remota como prueba de contenido o se puede prescindir de ella totalmente; también se pueden optar por imprimir las pruebas de color de las páginas ya impuestas en los pliegos.

Figura 9.1
Enlaces JDF/JMF
en preimpresión



La impresión de embalajes necesita un flujo de trabajo diferente. Para afrontar los múltiples procesos de preimpresión implicados, los flujos de trabajo se dividen en módulos. En la sección 4.3 ya vimos que PJTF utiliza la arquitectura de flujo de trabajo Extreme para comunicar el software de coordinación y los módulos del gestor de *Partes Electrónicas (Jobtickets)*. Esta función solo se hace cargo de JDF/JMF (flecha 2 figura 9.1), pero tanto en el caso de PJTF como en el flujo de trabajo JDF, el formato normalizado de intercambio de datos no es suficiente para que el usuario del WMS consiga la cooperación de los módulos instalados por distintos proveedores. Se necesitan acuerdos con todos los implicados.

9.1 Interfaz entre el MIS y Preimpresión

Antes de explorar la especificación *MIS a Preimpresión ICS*, trataremos ciertos aspectos que afectan directamente a este interfaz.

El sistema de gestión de órdenes debería permitirnos describir a la vez el trabajo y su ruta en preimpresión. Algunos sistemas WMS de preimpresión proponen flujos basados en la información JDF

Ejemplos de WMS compatibles con JDF

Fabricante:	Producto principal:
AC&C HSH Group	PuzzleFlow
Avato System GmbH	PMS Production Management System
Agfa	Apogee
DALiM	PRiNTEMPO
EFI	OneFlow
EskoArtwork	BackStage Server
Fujifilm	XMF
Krause-Biagosch GmbH	KIM
LithoTechnics Oty Limited	Metrix
Heidelberger Druckmaschinen AG	Prinect
Manroland	Printnet
Kodak	Prinergy
OneVision Software AG	Asura Pro
Rampage	Rampage Workflow System
Screen	TrueFlow
Xerox	FreeFlow Print Manager

Para más información, visite la **Matriz de Integración JDF**
<http://www.cip4.org/global/v3/index.php?content=matrix/intro.php>

que reciben del sistema gestor de órdenes/MIS. Pero ahora mismo desconocemos ciertos detalles relevantes para la producción que el propio WMS puede completar con valores por defecto. Por lo tanto, es evidente que las rutas de producción no están completamente definidas, aunque es algo que podemos cambiar.

En general, el software de preimpresión permite dividir la secuencia de trabajo en las siguientes etapas:

- 1 El administrador del sistema define nuevos flujos mediante secuencias de comandos (Scripting).
- 2 El Agente WMS de una imprenta define los valores por defecto para las rutas de trabajo y establece los parámetros de los productos. Estos ajustes complementan la base del flujo instalado por el proveedor.
- 3 El usuario adapta el flujo a sus productos basándose en las sugerencias del WMS.

La solución para programar secuencias de comandos suele proporcionarla el sistema operativo o el propio fabricante del WMS.

AppleScript [8] o **Windows Script Host WSH** [34] utilizan motores **VisualBasicScript** o **JavaScript**. El software **Prinergy** de Kodak utiliza la **automatización basada en normas (rules-based automation)**, perteneciente a la segunda categoría. Explicar estas tecnologías tan potentes sería largo y complejo. Si está interesado en la automatización planteada por Kodak (RBA), consulte el apartado de la página 126.

En el segundo caso, el agente del WMS puede ajustar por defecto en el RIP los valores de rasterizado o reventado (trapping). Los datos de contenido se pueden recibir en carpetas activas (*hotfolders*) creadas al efecto, imponer las páginas automáticamente en plantillas basándose en los nombres de archivo o incorporar máquinas nuevas al flujo de trabajo.

El tercer caso, muestra gráficamente los flujos del WMS en el monitor del usuario. Por ejemplo, este método permite desactivar el reventado o cambiar el valor de la trama por defecto del RIP.

El interfaz entre el MIS y preimpresión es especialmente crítico. Tras colocar un trabajo en el MIS, teóricamente, los detalles de preimpresión se trasladan al WMS y la orden cambia mientras comienza el trabajo. Algunos valores no suelen plantear demasiados problemas, como qué máquina realiza la impresión, el color o el número de copias (preimpresión puede grabar varios juegos de planchas para grandes tiradas).

De hecho, se puede actualizar el estado de la producción mediante el comando *actualizar JDF (UpdateJDF)* (envía el nuevo JDF al MIS para que no existan dos versiones). La actualización de documentos JDF será una cuestión clave resuelta en un futuro próximo.

En la especificación *MIS a Preimpresión ICS*, las Cajas Grises son creadas por el MIS principalmente como contenedores de información que debe ser completada con los datos suministrados por preimpresión. Existen 11 tipos de *Cajas Grises* en total:

- *Preparación de Preimpresión (PrepressPreparation)*.
- *Preparación de Imposición (ImpositionPreparation)*.
- *Prueba de Imposición (ImpositionProofing)*.
- *Rasterizado de la Imposición (ImpositionRIPing)*.

Automatización Mediante Normas (Rules-Based Automation RBA)

El flujo de trabajo Prinergy se puede cambiar y automatizar con las RBA en un interfaz externo al habitual del usuario. Se utiliza una herramienta gráfica específica para crear las normas.

Una norma, provoca un evento asociado a una acción.

Por ejemplo, si al “refinar” los datos (en argot Prinergy) o analizarlos se produce un error, la persona designada de la empresa recibe un mensaje por correo electrónico.

En el sistema Prinergy también se puede importar datos de contenido mediante carpetas activas, crear pruebas en segundo plano e intervenir manualmente en caso de fallo, tras recibir el aviso por correo electrónico.

Otro ejemplo: Utilizar nombres de archivos pactados para imponer automáticamente las páginas en la maqueta. Una norma RBA capta el evento “Relleno de Hoja” (Sheet Filled) y genera un pliego de prueba automáticamente.

Con las reglas RBA, también se puede integrar software ajeno al entorno Prinergy. Además, junto al lenguaje de programación Visual Basic se pueden usar las herramientas gráficas RBA para definir los controles del flujo de trabajo.

- *Ajuste de Planchas (PlateSetting).*
- *Grabado de Planchas (PlateMaking).*
- *Pruebas de Pantalla de la Imposición (ImpositionSoftProofing).*
- *Prueba de Página (PageProofing).*
- *Pruebas de Pantalla de Página (PageSoftProofing).*
- *Creación del Contenido (ContentCreation).*
- *Elaboración de Pruebas y Planchas (ProofAndPlateMaking).*

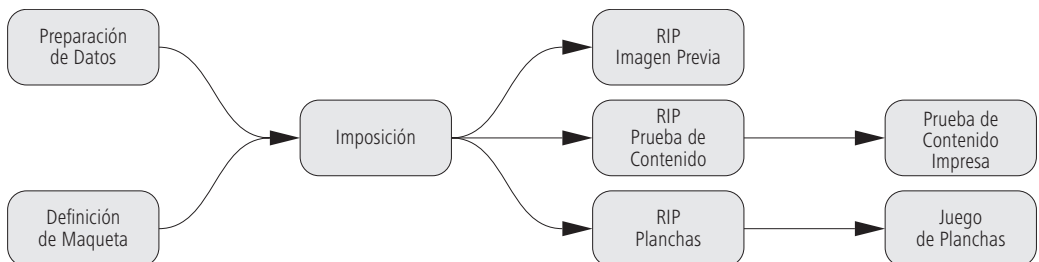
Para describir el flujo de trabajo de preimpresión, las *Cajas Grises* se pueden ensamblar como si fueran bloques de construcción pero, de igual modo que en la vida real, estos pueden ser grandes o pequeños.

Aclaremos esta confusa analogía basándonos en la figura 9.2. Aquí representamos el flujo de trabajo típico realizado por un RIP (imagen previa, prueba de contenido y planchas, etc). Entre los procesos expuestos pueden haber grandes diferencias: la resolución de una imagen previa en pantalla es 72 ppp, la prueba de contenido puede llegar a 600 ppp y las planchas del CTP necesitan unos 2400 o 2540 ppp. Además según el uso previsto, la imagen se necesita compuesta o separada. Las pruebas de contenido requieren imágenes compuestas y las planchas separaciones de color.

Las actividades de la figura 9.2 son otro ejemplo de *Cajas Grises*:

- Preparación de Datos para Preimpresión (**PrepressPreparation**).
- Definición de la Maqueta de Imposición (**ImpositionPreparation**).
- Imposición, RIP y Prueba de Contenido (**ImpositionProofing**).

Figura 9.2
Diagrama de flujo de las actividades de Preimpresión para impresión comercial

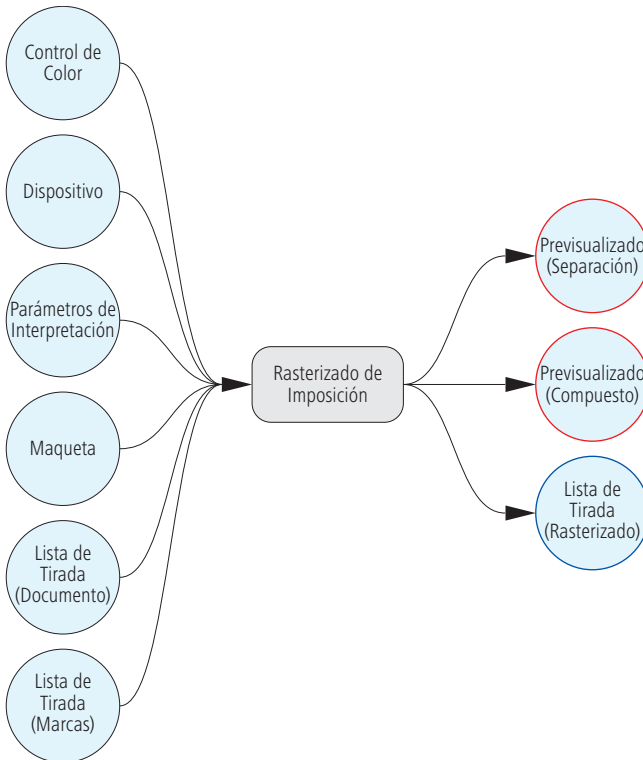


- Imposición, RIP de Planchas y RIP de la vista previa opcional (**ImpositionRIPing**).
- Grabado de Planchas (**PlateSetting**).

La Caja Gris *Grabado de Planchas* es un gran bloque de construcción que combina *Rasterizado de Imposición* y *Ajuste de Planchas*, provocando que las dos últimas sean redundantes. La *Caja Gris* combinada *Prueba y Grabado de Planchas* es otra *Caja Gris* mayor, se compone de *Prueba de Imposición* y *Grabado de Planchas*.

La especificación *MIS a Preimpresión ICS* define un atributo por cada *Caja Gris*, pero sobre todo especifica los recursos necesarios de entrada y salida y sus atributos. En las figuras 9.3 a 9.6 vemos las *Cajas Grises Rasterizado de Imposición*, *Ajuste de Planchas* y *Grabado de Planchas* con sus recursos, esta última ilustrada en la figura 6.18. Habitualmente, *Imposición* y *Ajuste de Planchas* están encadenadas y la transferencia entre ambos es el recurso circular con borde azul *Tirada (Datos Ripeados)* que representa las separaciones tramadas en TIFF. En la figura 9.6 se muestra la composición de las *Cajas Grises Grabado de Planchas*, *Rasterizado de Imposición* y *Ajuste de Planchas*.

Figura 9.3
Rasterizado de Imposición
mediante Cajas Grises



En este caso, la salida de *Rasterizado de Imposición* no sirve como entrada para *Ajuste de Planchas*, con la salida de *Grabado de Planchas* ocurre lo mismo. Hemos marcado los *Recursos de Previsualización* con un borde rojo para aclararlo. Además, los recursos de entrada *Rasterizado de Imposición* y *Ajuste de Planchas* deben incluirse para la *Producción de Planchas*; el resultado intermedio es el objeto con borde azul *Lista de Tirada (Rasterizado)*.

Un sistema MIS puede describir el flujo de preimpresión de un modo diferente, como muestra la figura 9.6 donde las flechas simbolizan los resultados de las *Cajas Grises*.

El MIS o Administrador puede construir las *Cajas Grises* de distintas formas y el flujo de trabajo de preimpresión o *Trabajador* debe ser capaz de interpretar todas las versiones de las mismas para transformarlas en nodos de proceso JDF durante el transcurso de la producción.

La norma *MIS a Preimpresión ICS* también establece niveles graduales de implementación. Hasta ahora, todas las *Cajas Grises* son de Nivel 1 y cumplen el Nivel 2 automáticamente.

El Nivel 2 debe cumplir normas adicionales sobre el control de las versiones: múltiples lenguajes usados en un producto o la capacidad de realizar productos complejos (por ejemplo, cubierta y contenido) donde se unen diferentes partes para realizar un solo documento.

Por último, si la especificación *MIS a Preimpresión ICS* está basada en otras normas ICS debe cumplir todos los requisitos de estas.

También podemos afirmar que para las normas *Base ICS* y *JMF ICS* las 2 fases del *MIS a Preimpresión ICS* son de Nivel 2 aunque los pasos son diferentes.

Obviamente, un requisito del *MIS a Preimpresión ICS* es cumplir la norma *ICS MIS*.

Cumplir los Niveles 1 y 2 *ICS MIS* son requisitos previos de Nivel 2 del *MIS a Preimpresión ICS*.

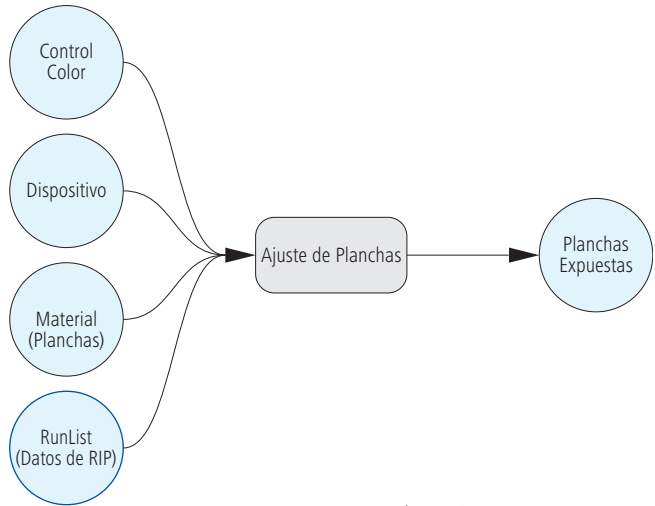


Figura 9.4
Caja Gris Ajuste de Planchas

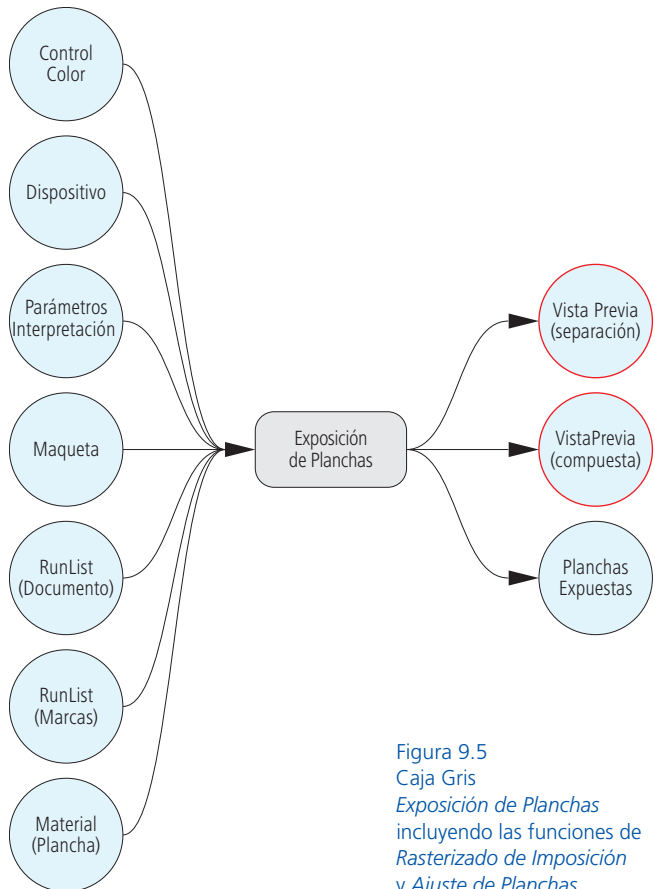


Figura 9.5
Caja Gris
Exposición de Planchas
incluyendo las funciones de
Rasterizado de Imposición
y Ajuste de Planchas

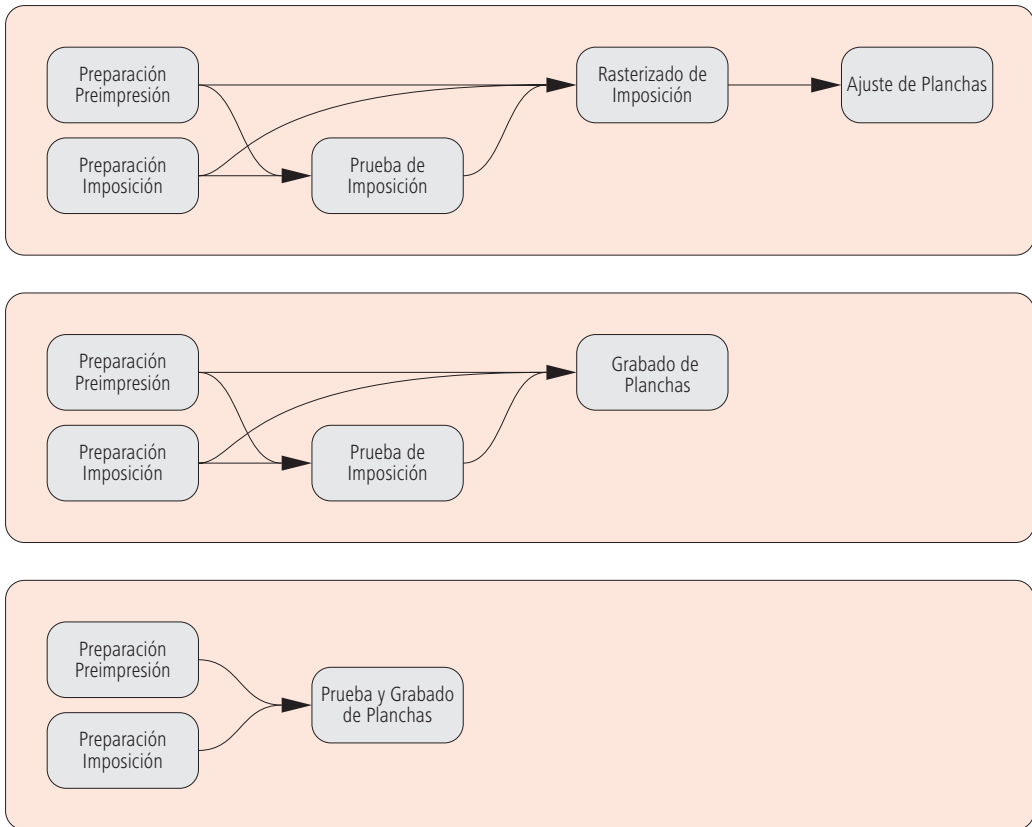


Figura 9.6
Tres formas distintas de
enlazar Cajas Grises

9.2 Montaje

El montaje consta de tres fases sucesivas:

- Creación, maquetación e imposición.
- Ajustes de espacios y medidas de la plantilla de imposición.
- Imposición, cálculo para distribuir los datos de impresión en las planchas. También se incluyen las páginas individuales y marcas (habituales en estructuras PDF).

Los tres primeros pasos se suelen realizar en el programa de maquetación. Es bastante común hacer el montaje en el mismo programa, pero las asignaciones y la imposición, las realiza el potente servidor del flujo de trabajo WMS en segundo plano. Sin embargo, la creciente integración de maquetación, montaje y el software WMS está diluyendo la división entre los programas.

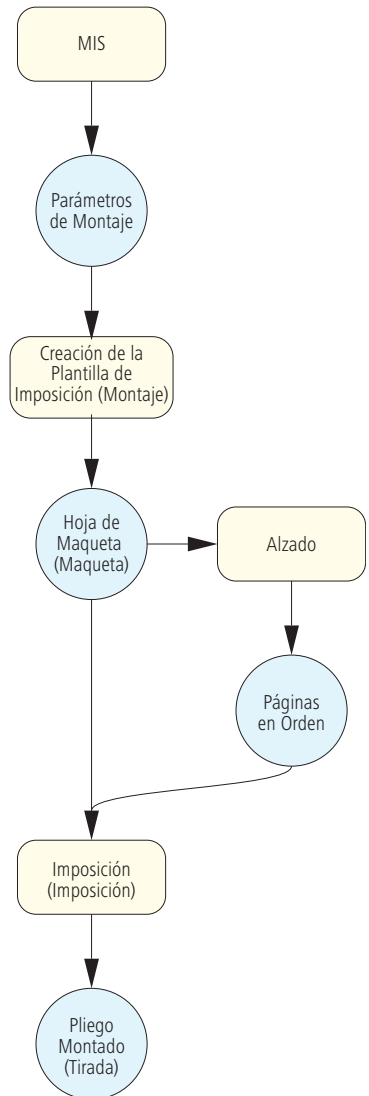
Esta sección trata la creación de las plantillas de imposición denominada en lenguaje coloquial plantilla de montaje. Las tareas habituales son:

- Situación de la hoja de impresión sobre la plancha.
- Definición de los tamaños de página y hoja.
- Ajuste de márgenes y espaciado (pinza, lomo, guillotina, etc.)
- Inserción de los elementos de control en planchas y pliegos (tiras de control/color, marcas de registro, plegado, corte, página, alzado, pliego, etc).

Figura 9.7
Modelo de recursos del proceso de montaje

Gran parte de la información es proporcionada por el MIS. En el capítulo anterior analizamos los *Parámetros de Montaje* (*StrippingParams*) proveedores de estos datos. La figura 9.7 muestra la estructura simplificada. Nos gustaría recalcar que los recursos y procesos del flujo de trabajo son tareas generalizadas, no representan estructuras JDF.

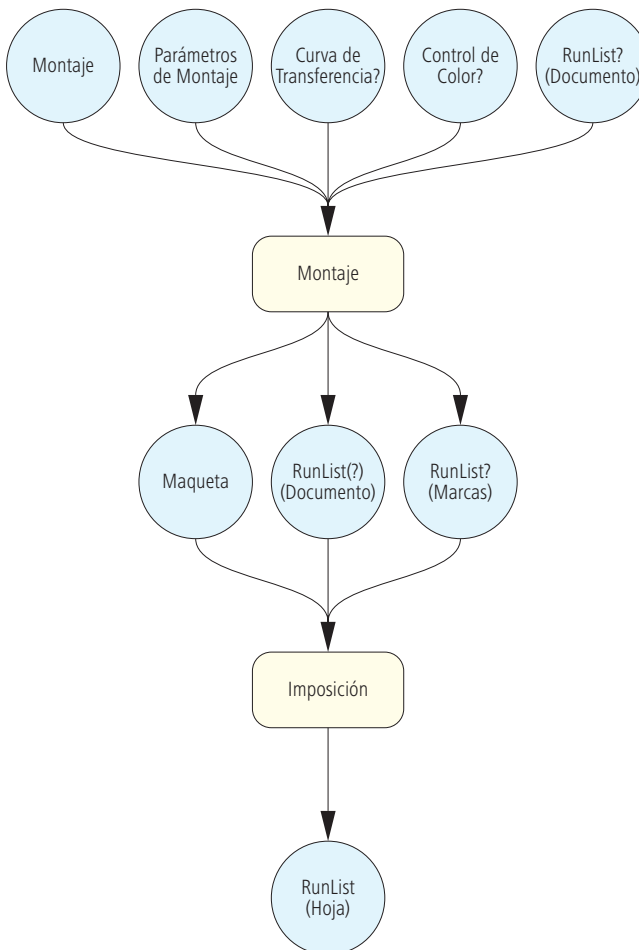
La secuencia tiene potencial para la automatización. Si no existe conexión JDF con el sistema gestor de pedidos todas las especificaciones del pliego se deben introducir manualmente en el programa de montaje cada vez que necesitamos una nueva imposición. Obviamente, podríamos guardar las maquetas de imposición para utilizarlas más tarde y alterar pequeños detalles para adaptarlas a órdenes similares. Sin embargo, todos estos valores se podrían incorporar directamente y generar el diseño de imposición de forma automática, siempre que el sistema gestor de pedidos y preimpresión estuvieran integrados mediante JDF. En el mundo del offset seguramente sería necesario cambiar ciertas cosas de la maqueta, pero un operador podría encargarse de ello. En este caso, el requisito previo es que las marcas de control se coloquen según el tamaño del pliego y no por coordenadas fijas. Los pliegos de distintos tamaños necesitan situar las marcas automáticamente en lugares diferentes y dichas marcas no están en el sistema de gestor de pedidos porque allí solo se inserta el tamaño. En la actualidad este proceso se define en el software de montaje. Desde JDF 1.4 se pueden definir marcas dinámicas directamente en el MIS con el recurso *StrippingParams*, sin que el proceso de montaje se haya realizado todavía.



Sin embargo, en el momento de generar el pedido es necesario que todos los requisitos estén cumplimentados, tanto los que están a cargo del personal como los correspondientes al software.

Ahora analizaremos la estructura JDF de la figura 9.7. En la figura 8.7 ya hemos visto la creación del esquema de montaje o imposición con sus recursos de entrada y salida, la figura 9.8 lo muestra detalladamente. El símbolo “?” significa que los recursos son opcionales. Como podemos ver en el proceso de montaje los recursos *Curva de Transferencia*, *Control de Color* y *Lista de Ejecución (Documento)* ya se han integrado. Las curvas de calibración están en el recurso *Curva de Transferencia*, estos valores tonales solo son importantes para el RIP y las trataremos en la sección 9.4. Este recurso se utiliza para transformar la información entre imposición y planchas cuando es realmente necesario.

Figura 9.8
Montaje en el modelo JDF



Los colores de separación se configuran en *Control de Color* para que las tiras de control de color correspondientes se generen simultáneamente en la maqueta de imposición.

En el proceso de *Montaje*, el documento definido por el recurso *RunList* puede cambiar al importar las páginas, entonces los datos de imposición se pueden controlar o reconstruir visualmente en pantalla siempre que esta función esté disponible. Si es este el caso, el recurso *Tirada* se puede trasladar de nuevo al proceso *Imposición*. El símbolo “?” significa que pueden pero no debe ser recurso de salida en el proceso de *Montaje*. De cualquier forma el recurso *Tirada* debe ser siempre recurso de entrada en un proceso de *Imposición* sea cual sea su procedencia, de lo contrario no habría nada que imponer. Por eso el símbolo de interrogación está entre paréntesis (?).

Habitualmente, el proceso de *Montaje* genera sobre la plancha uno o varios archivos correctamente situados con sus marcas de control descritas por el recurso *Tirada (Marcas)*. En realidad JDF solo puede posicionar las marcas de control ya que su apariencia se puede establecer, por ejemplo, en un PDF aparte.

Para finalizar, veamos el recurso *Maquetación (Layout)* de la figura 9.9, es un *marcador de posición (placeholder)* de 6 páginas. En lenguaje JDF placeholder es un *Objeto de Contenido (ContentObject)*. La posición de las marcas es distinta para cada signatura, hoja, frontal o reverso. La maqueta se divide según indica los recursos *Signatura (SignatureName)* y *Cara (SheetNameSide)*. En la figura 9.10 vemos la maquetación del pliego. El valor *Contenido por Cara (SurfaceContentsBox)* es el mismo que *Tamaño (Dimension)* de la *Plancha (Media)*. Generalmente el valor de *SurfaceContentsBox* puede ser menor, es el pequeño rectángulo que incluye todos los elementos de impresión en el pliego (= *BoundingBox*).

Los atributos del elemento *ContentObject* son los siguientes:

CTM:

En el capítulo 4.3 se introdujo la *Matriz Actual de Transformación CTM* como entrada del *Diccionario PDF*. Este concepto original de PostScript tiene una función parecida en JDF. Define la posición, orientación y tamaño de los marcadores de posición en *SurfaceContentsBox* y sobre la plancha (ver ejercicio).

ClipBox:

Caja de Corte. En el software de maquetación se establece el tamaño y posición de un marcador de posición en puntos DTP. Los elementos de página fuera de *ClipBox* se cortan. La posición hace referencia a *SurfaceContentsBox*.

PositionX y PositionY:

(*PosiciónX*, *PosiciónY*). Coordenadas del marcador de posición. Suele ser la orientación del contenido.

TrimSize:

Tamaño de la *Caja de Corte* en puntos DTP. Suelen ser las páginas tras el corte o el formato del producto final impreso.

TrimCTM

Posicionamiento de la *Caja de Corte (TrimBox)* en la *Caja de Contenidos (SurfaceContentsBox)*

```

<Layout Class="Parameter" DescriptiveName="Status" ID="_300"
PartIDKeys="SignatureName SheetName Side" Status="Available">
  <Layout Name="Signature-1" SignatureName="Signature-1">
    <Layout DescriptiveName="Cover" Name="Cover" SheetName="Cover"
SourceWorkStyle="WorkAndBack"
SurfaceContentsBox="0 0 2111.81 1714.96">
      <Media Class="Consumable" Dimension="2111.81 1714.96"
MediaType="Plate" />
      <Layout DescriptiveName="Recto" Side="Front">
        <ContentObject CTM="1 0 0 1 316.06 188.50"
ClipBox="307.55 180 1036.06 "
PositionX="Center" PositionY="Center"
TrimCTM="1 0 0 1 316.06 188.50" TrimSize="711.49 340.15" />
        <ContentObject CTM="1 0 0 1 1084.25 188.50"
ClipBox="1075.74 180 1804.25 537.16"
PositionX="Center" PositionY="Center"
TrimCTM="1 0 0 1 1084.25 188.50" TrimSize="711.49 340.15" />
        <ContentObject CTM="1 0 0 1 316.06 585.35"
ClipBox="307.55 576.85 1036.06 934.01"
PositionX="Center" PositionY="Center"
TrimCTM="1 0 0 1 316.06 585.35" TrimSize="711.49 340.15" />
        <ContentObject CTM="1 0 0 1 1084.25 585.35"
ClipBox="1075.74 576.85 1804.25 934.01"
PositionX="Center" PositionY="Center"
TrimCTM="1 0 0 1 1084.25 585.35" TrimSize="711.49 340.15" />
        <ContentObject CTM="1 0 0 1 316.06 982.20"
ClipBox="307.55 973.70 1036.06 1330.86"
PositionX="Center" PositionY="Center"
TrimCTM="1 0 0 1 316.06 982.20" TrimSize="711.49 340.15" />
        <ContentObject CTM="1 0 0 1 1084.25 982.20"
ClipBox="1075.74 973.70 1804.25 1330.86"
PositionX="Center" PositionY="Center"
TrimCTM="1 0 0 1 1084.25 982.20" TrimSize="711.49 340.15" />
        <MarkObject CTM="1 0 0 1 0 0" ClipBox="0 0 2111.81 1714.96">
          <RegisterMark Center="1900.86 661.94" Class="Parameter"
Rotation="0" />
          <RegisterMark Center="210.94 661.94" Class="Parameter"
Rotation="0" />
          <ColorControlStrip Center="1059.44 1347.70"
Class="Parameter" Rotation="0" Size="1486.18 28"
StripType="FOGRA_6_F74_740x10" />
        </MarkObject>
      </Layout>
    </Layout>
    ...
  </Layout>
</Layout>

```

Figura 9.9
Código del recurso Layout

En el recurso *Objetos de Marca (MarkObject)* se han insertado la posición, la tira de control "FOGRA 6 F74 740x10" y las marcas de registro.

En las diferentes versiones de la especificación JDF, las modificaciones del recurso *Maqueta (Layout)* son muy importantes y debemos aclarar que su apariencia en esta versión es muy distinta a la de anteriores versiones JDF.

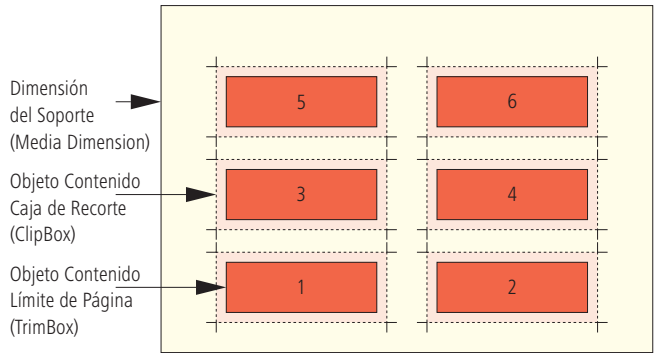


Figura 9.10 Límites de los objetos. Atributos de ContentObject de la página final (TrimSize) y marcas de corte (ClipBox).

9.3 Sobreimpresión y Reventado (Trapping)

La sobreimpresión de los distintos colores utilizados en impresión offset puede producir colores desalineados. Estas variaciones de registro se producen, básicamente, por estiramiento del papel o tolerancias de la prensa.

En la figura 9.11 mostramos un "error de registro". Para resolver estos pequeños errores de registro se amplían las áreas adyacentes (solapado). Exponemos la norma utilizada habitualmente en la figura 9.12, donde los colores translúcidos se superponen a los más oscuros.

Otro aspecto a considerar son los contornos que pueden aparecer al excedernos en el control del *Trapping*. En este sentido, el valor del reventado debe ser igual a la diferencia máxima de registro. Solo podemos utilizar un valor mayor cuando el objeto a sobreimprimir sea negro. Al usar colores opacos, como oro o plata, la norma expuesta anteriormente no es aplicable.

Un color translúcido no debe superponerse a otro opaco, obviamente debe ser al contrario. Al imprimir dos colores opacos adyacentes, el primero solapa al otro.

Los valores del *Trapping* se deben especificar porque no existen ajustes por defecto ni reglas universales.



Figura 9.11 Desajuste causado por variaciones de registro sin aplicación de trapping

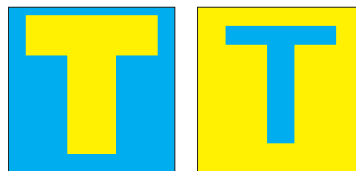


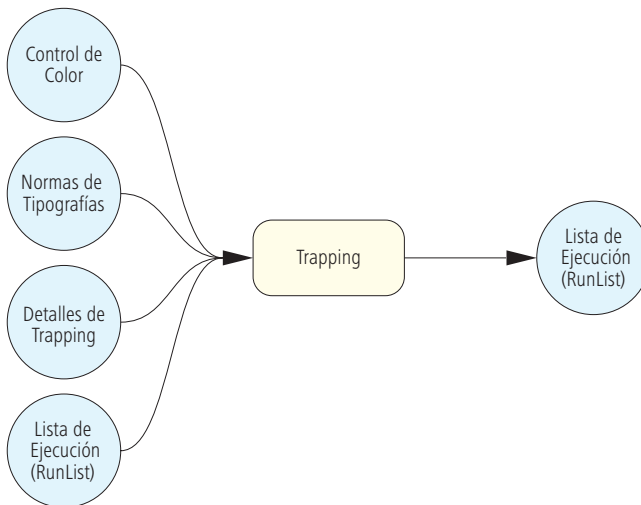
Figura 9.12 Los colores más claros se solapan sobre los oscuros

Los ajustes dependen de diferentes factores (el tipo de papel, la prensa utilizada, técnica de impresión, etc) y se definen en *Detalles de Trapping* (*TrappingDetails*), recurso de entrada del proceso (figura 9.13). En *Control de Color* (*ColorantControl*) se especifica si los colores son translúcidos u opacos.

Cuando falta un tipo de letra, el recurso *Normas de Tipografías* (*FontPolicy*) determina la acción a realizar: detener, seleccionar fuente alternativa, etc. Finalmente los datos de contenido para impresión están en *Lista de Ejecución* (*RunList*), una vez incluidos los datos de *Trapping* obtenemos un nuevo recurso *Lista de Ejecución*.

En la figura 9.14 vemos el ejemplo de recurso *TrappingDetails* preparado para el motor de *Trapping*. Es prácticamente un RIP y el interfaz gráfico del motor de *Trapping* en preimpresión es una patente del proveedor y no se puede configurar fuera del WMS aunque esté basado en JDF.

Figura 9.13
Recursos del proceso de
sobreimpresión



Ahora explicaremos los elementos y atributos mostrados en el código de la figura 9.14. Si *Trapping por Defecto* es *Cierto* (*Default Trapping = "True"*) la sobreimpresión se aplica a la página entera, si es *"False"* se aplica a zonas determinadas. La superposición de los colores, sobre todo de los opacos es muy importante, la secuencia de impresión también y esta se especifica en el subelemento *Orden de Trapping* (*TrappingOrder*).

El recurso *Parámetros de Trapping* (*TrappingParams*) alberga muchos detalles de sobreimpresión. A continuación explicamos algunos de ellos.

El atributo *Límite de Color Negro* (*BlackColorLimit-95%*) se rige por la norma del color negro comentada al inicio de la sección 9.3 y es el valor tonal que sobreimprime.

Densidad Máxima de Negro (*BlackDensityLimit*) indica el valor de densidad de color considerada análoga a negro sólido.

El atributo *Anchura de Negro* (*BlackWidth*) establece el ancho de reventado del negro. Su valor es 1,3 veces la anchura de reventado de los demás colores.

Sobreimpresión Imagen a Imagen (*ImageToImageTrapping*) y *Sobreimpresión Imagen a Objeto* (*ImageToObjectTrapping*) definen la sobreimpresión de las imágenes si se superponen o son colindantes a objetos gráficos, textos, etc. El valor *True* indica que los límites entre imágenes u objetos se sobreimprimen.

Posición Trapping de Imagen (*ImageTrapPlacement*) está indicado como "Normal" y se aplicará la regla habitual de luminosidad para colores translúcidos y la secuencia de impresión indicada para colores opacos.

También se puede *Sobreimprimir* el objeto sobre la imagen mediante el valor "Choke". Como vemos en la figura 9.12, una separación de color tiene suficiente diferencia tonal para crear la sobreimpresión. Los límites para la sobreimpresión de degradados se establecen en *StepsLimit*. Por último, en *TrapWidth* se define el *Ancho de Reventado* en puntos DTP.

9.4 Rasterizado y Producción de Planchas

En la figura 6.17 el sistema de producción dividía las *Cajas Grises* creadas por el MIS en procesos individuales y en las figuras 9.4 y 9.5 desglosamos *Rasterizado de Imposición, Ajuste y Exposición de Planchas* (*ImpositionRIPing, PlateMaking* y *PlateSetting*).

Figura 9.14
Fragmento del recurso
Detalles de Reventado

```
<TrappingDetails Class="Parameter" DefaultTrapping="true" ID="_333"
  Locked="false" Status="Available">
  <TrappingOrder>
    <SeparationSpec Name="Black" />
    <SeparationSpec Name="Cyan" />
    <SeparationSpec Name="Magenta" />
    <SeparationSpec Name="Yellow" />
    <SeparationSpec Name="Gold-Spot color" />
  </TrappingOrder>
  <TrappingParams
    BlackColorLimit="0.95"
    BlackDensityLimit="1.6"
    BlackWidth="1.3"
    ImageToImageTrapping="false"
    ImageToObjectTrapping="true"
    ImageTrapPlacement="Normal"
    StepLimit="0.25"
    TrapWidth="0.25"
    ... />
</TrappingDetails>
```

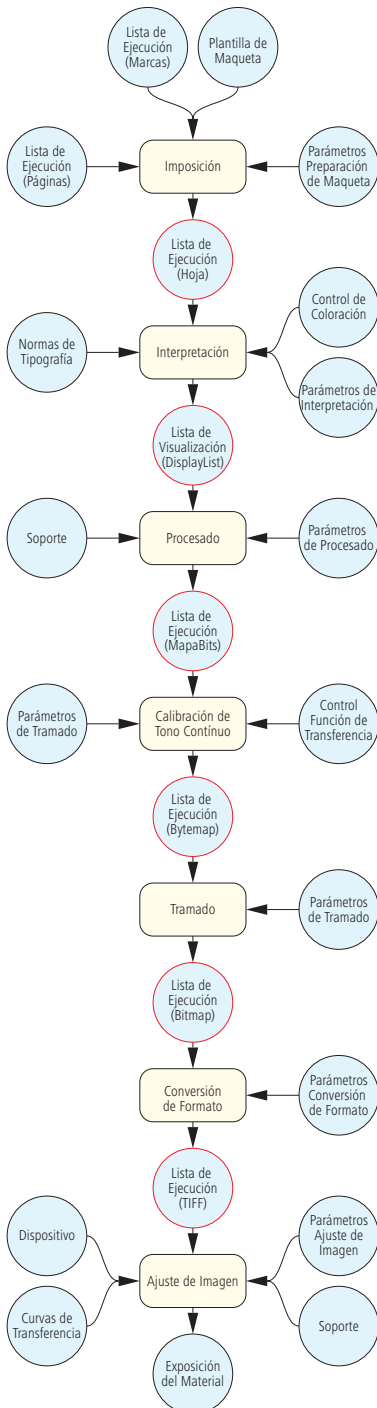


Figura 9.15

Modelo JDF para exposición de planchas

Los tres procesos se pueden combinar pero debemos plantearlo como la resolución de un puzzle encajando el recurso de salida de un proceso en la entrada del siguiente y así sucesivamente; la mayoría se gestionan mediante recursos *Lista de Ejecución (RunList)* y sus resultados intermedios suelen ser datos aunque algunos procesos combinados pueden extraviarlos. Conocemos algunas piezas del puzzle como *Interpretación*, *Procesado* y *Tramado* (ver figura 3.10). *Imposición* y *Ajuste de Imagen* ya se mencionaron en la sección 6.1 pero hemos considerado conveniente ampliarlas en la figura 9.15 mostrando los detalles estructurales del procesado RIP y la exposición de planchas.

Calibración de Tono Continuo (ContoneCalibration) es el proceso que aplica la curva de transferencia en el RIP. Cada separación es un mapa de bits procedente del *Procesado (Rendering)*, por tanto, cada color CMYK o tinta especial contiene los valores tonales del 0% al 100%. El resultado es una estructura idéntica con los valores tonales modificados y los cambios que se almacenan en el recurso *Control de Transferencia (TransferFunctionControl)*.

Dichas curvas de transferencia se describen de forma similar a las del PPF mostrado en la figura 4.11. De hecho, por cada separación podrían utilizarse más curvas. La linealización se suele realizar con una función de transferencia que ajusta el valor tonal de la plancha para que la diferencia entre valores de entrada del RIP y plancha revelada sean mínimos.

Aparte de la curva de linealización de las planchas, está la curva real de la prensa. Depende de sus propias condiciones y parámetros, como la trama de salida, el soporte y los colores. Ambas curvas se calculan en conjunto.

Con el proceso *Conversión de Formato* se transforman los datos y los mapas de bits específicos del proveedor convertidos a TIFF (TIFF-B" o tiff 1bit) son utilizados por el software del CTP como recursos de entrada. El tiff 1bit es un mapa de píxeles en blanco y negro sin tonos grises conocido como datos binarios. Prácticamente, es el estándar de los flujos de preimpresión lo que permite al WMS conectar con distintas filmadoras. El resultado es que la exposición de las planchas suele ser un proceso separado del WMS.


```

<JDF Activation="Active" ID="_001" JobPartID="_1003.I" Status="Part"
  Type="Combined" Types="Imposition Interpreting Rendering ContoneCalibration
  Screening PreviewGeneration FormatConversion ImageSetting" Version="1.3">
...
  <ResourceLinkPool>
    <RunListLink CombinedProcessIndex="0" ProcessUsage="Document"
      Usage="Input" rRef="_77" />
    <RunListLink CombinedProcessIndex="0" ProcessUsage="Marks"
      Usage="Input" rRef="_21" />
    <LayoutPreparationParamsLink CombinedProcessIndex="0" Usage="Input"
      rRef="_581" />
    <LayoutLink CombinedProcessIndex="0" Usage="Input" rRef="_99" />
    <ColorantControlLink CombinedProcessIndex="1" Usage="Input"
      rRef="_83" />
    <FontPolicyLink CombinedProcessIndex="1" Usage="Input" rRef="_583" />
    <TransferCurvePoolLink CombinedProcessIndex="6" Usage="Input"
      rRef="_71" />
    <InterpretingParamsLink CombinedProcessIndex="1" Usage="Input"
      rRef="_85" />
    <RenderingParamsLink CombinedProcessIndex="2" Usage="Input"
      rRef="_86" />
    <MediaLink CombinedProcessIndex="2" ProcessUsage="Paper" Usage="Input"
      rRef="_79" />
    <ScreeningParamsLink CombinedProcessIndex="3 4" Usage="Input"
      rRef="_89" />
    <TransferFunctionControlLink CombinedProcessIndex="3" Usage="Input"
      rRef="_34" />
    <ImageSetterParamsLink CombinedProcessIndex="6" Usage="Input"
      rRef="_94" />
    <DeviceLink CombinedProcessIndex="6" Usage="Input" rRef="_286" />
    <MediaLink CombinedProcessIndex="6" ProcessUsage="Plate" Usage="Input"
      rRef="_3085" />
    <FormatConversionParamsLink CombinedProcessIndex="5" Usage="Input"
      rRef="_47" />
    <ExposedMediaLink CombinedProcessIndex="6" ProcessUsage="ExposedMedia"
      Usage="Output" rRef="_81" />
  </ResourceLinkPool>
...
</JDF>

```

Generalmente, la filmadora/CTP y su software no soporta JDF/JMF. En estos casos el flujo de trabajo JDF genera un TIFF-B simultáneo a la finalización de las planchas y emite el mensaje oportuno. La especificación *MIS a Preimpresión ICS* contempla esta función explícitamente.

El código de la figura 9.16 es el proceso combinado de la 9.15, sin embargo, solo está el *Fondo de Enlace de Recursos* (*ResourceLinkPool*); *Fondo de Recursos* (*ResourcePool*) no. Vemos que los recursos *RunList* en rojo de la figura 9.15 no se encuentran ni los enlaces de *ResourceLinkPool* ni los *ResourcePool* permitidos.

Figura 9.16
Proceso combinado para
exposición de planchas

```
<FormatConversionParams Class="Parameter" ID="_47" Status="Available">
  <FileSpec Class="Parameter" MimeType="application/octet-stream"
    ResourceUsage="InputFormat" />
  <FileSpec Class="Parameter" MimeType="image/tiff"
    ResourceUsage="OutputFormat" />
</FormatConversionParams>
```

Figura 9.17
Recurso Parámetros para
Conversión de Formato

La figura 9.17 muestra el recurso *Parámetros de Conversión de Formato* (*FormatConversionParams*) a modo de ejemplo donde una aplicación especificada produce una imagen TIFF partiendo de la secuencia de bytes de un proceso combinado.

Para explicar el concepto de impresión digital se puede utilizar un modelo JDF similar, pero en la práctica lo habitual es enviar los pedidos directamente al software controlador de la prensa digital donde un operador puede intervenir para ordenarlos, según el tipo de papel por ejemplo.

Para finalizar esta sección queremos recordar que la comunicación entre JDF, RIP, WMS y filmadora/CTP suele estar restringida por el fabricante, este utiliza JDF para controlar sus módulos y evitar la intrusión de programas ajenos. Entre otras opciones también puede usar bases de datos y es legítimo en ambos casos; el resultado de nuestra investigación no es sorprendente y en determinadas circunstancias JDF se utiliza de un modo "creativo". Hay procesos combinados que no precisan resultados intermedios y no tiene sentido incluirlos en recursos *Lista de Ejecución* porque el WMS los memoriza en carpetas, bases de datos o en la memoria RAM.

9.5 Pruebas y Aprobaciones de Prensa

Una prueba debe representar por adelantado los resultados de la tirada en impresión y se realiza con los siguientes métodos:

- Prueba de color vinculante o Prueba de contrato (digital).
- Prueba de imposición o contenido. En argot, prueba ferro.
- Prueba electrónica de pantalla (Soft Proof).
- Pruebas tramadas.
- Prueba de contacto o de máquina (habitual en hueco y flexo).
- Prueba analógica (obtenida de fotolitos).

Las tres primeras tecnologías juegan un papel crucial en la actualidad, mientras que las dos últimas cada vez se utilizan menos.

Las pruebas son parte esencial del control de calidad donde también se analizan otros elementos importantes como las planchas, los pliegos u otros componentes del producto final. Los objetivos principales son: la verificación del operario para decidir la continuidad del proceso o el permiso del cliente para imprimir el trabajo.

A continuación incluimos los recursos JDF para admitir o desautorizar la producción. *Soporte Expuesto (ExposedMedia)* se usa para la prueba impresa de contenido o color y los recursos *Lista de Ejecución (RunList)* se utilizan para los datos de imagen de pruebas de pantalla.

La figura 9.18 ilustra el caso. Los recursos son *Parámetros de Aprobación (ApprovalParams)* y la información sobre la persona que inicia el proceso (o la autorización a otra persona) está en el subelemento *Contacto*. La salida aprobada o rechazada se incluye en un proceso *Aprobación (Approval)*. Antes de este el valor del atributo *Estado (Status)* es en *Proyecto (Draft)*. Otros estados posibles son *Aprobado (Available)* o *Rechazado (Rejected)*.

A continuación se indica quien aprobó o rechazó la prueba en el recurso *Aprobado con éxito (ApprovalSuccess)* y se inserta un enlace al fichero con su firma y los comentarios posibles, si procede.

El proceso *Approval* se puede definir como el estado final de un proceso combinado. La prueba impresa en una imprenta digital es un proceso combinado que finaliza con los procesos contenidos en *Aprobación de Impresión Digital (DigitalPrintingApproval)* como vemos en el código de la figura 9.19. Para realizar una prueba de máquina offset no se utiliza el mismo proceso combinado, se usan los procesos *Impresión Convencional (ConventionalPrinting)* y *Ajustes de Imagen (ImageSetting)*. En el siguiente capítulo detallamos *ConventionalPrinting* y *DigitalPrinting*.

Para la obtención de pruebas de color y la exposición de planchas es necesario realizar transformaciones entre los distintos espacios de color.

Los perfiles ICC describen las características de color de cada dispositivo.

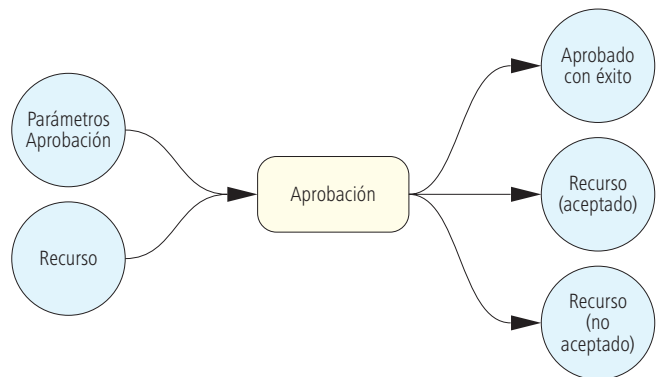


Figura 9.18
Proceso de aprobación JDF

Por regla general necesitamos realizar conversiones de color para conseguir pruebas de color, planchas e impresiones en prensa y estas transformaciones se realizan con perfiles de color ICC que representan las características de color de cada dispositivo. Para independizar el color de los datos procedentes de cámaras digitales o escáneres, necesitamos un perfil RGB del dispositivo y un perfil de impresión, que puede ser RGB o CMYK según la tecnología de salida utilizada para imprimir los documentos. Las pruebas de color requieren un perfil del dispositivo habitualmente denominado perfil de prueba.

Figura 9.19
Proceso de aprobación en
Impresión Digital

```
<JDF Category="DigitalPrinting" ID="_1" JobID="_2" Status="Ready"
  Type="Combined" Types="LayoutPreparation Imposition Interpreting Rendering
  ImageSetting Approval"...>
...
</JDF>
```

Las transformaciones entre espacios de color en el JDF se realizan antes del proceso de *Trapping* mediante el proceso *Conversión entre Espacios de Color* (*ColorSpaceConversion*). La ubicación de los perfiles está en el recurso de entrada *Parámetros de Conversión de Espacio de Color* (*ColorSpaceConversionParams*); este es el recurso más importante de dicho proceso.

En la figura 9.20 podemos ver su estructura y elementos y atributos que desglosamos a continuación:

Figura 9.20
Información para
transformación del
espacio de color en
prueba de pantalla

```
<ColorSpaceConversionParams ColorManagementSystem="ADBE" ID="R6"
  Class="Parameter" Status="Available">
  <ColorSpaceConversionOp SourceCS="Gray" SourceObjects="All"
    Operation="Untag" />
  <ColorSpaceConversionOp SourceCS="RGB" SourceObjects="All"
    IgnoreEmbeddedICC="false" RGBGray2Black="true" RGBGray2BlackThreshold="1"
    Operation="Tag" RenderingIntent="Perceptual" >
    <FileSpec ResourceUsage="SourceProfile"
      URL="file://Server1/ICCProfiles/sRGBprofile.icm" />
  </ColorSpaceConversionOp>
  <ColorSpaceConversionOp SourceCS="CMYK" SourceObjects="All"
    Operation="Untag" />
  <FileSpec ResourceUsage="FinalTargetDevice"
    URL="file://Server1/ICCProfiles/CoatedFOGRA39.icc" />
</ColorSpaceConversionParams>
```

Sistema de Gestión de Color (ColorManagementSystem):

Espacio de conversión de color especificado por el proveedor del sistema. En este caso "Adobe".

Especificación de Archivo (FileSpec):

Este atributo define el objetivo de color con los valores del perfil ICC incluido ("sRGBprofile.icm").

Opciones de Conversión (ColorSpaceConversionOp):

Opciones de conversión entre espacios de color:

- *SourceCS*. *Espacio de Color de Origen* del modelo de color.
- *SourceObject*. El *Tipo de Objeto* que se transforma, como *Text*, *LineArt*, *ImagePhotographic*, etc. Si se aplica a todos los objetos se define como *(All)*.
- *IgnoreEmbeddedICC*. Si los perfiles incrustados se ignoran o no se define mediante *true* o *false*.
- *RGBGray2Black*. *True* o *false* especifica si los *grises RGB* se convierten al canal negro con conversión CMYK o no. En este caso si se convierten.
- *RGBGray2BlackThreshold*. *Umbral de conversión Gris RGB a Negro*. Atributo que limita la aplicación del valor tonal de conversión de RGB a CMYK.
0 = Solo el negro RGB es tono completo.
0,5 = Todos los grises RGB son 50%.
1 = Todos los grises RGB se asignan al canal negro.
- *RenderingIntent*. *Propósito de Conversión* que determina el mapa de gama utilizado para trasladar los colores desde un espacio de color a otro. La transformación visual del ejemplo describe que se ha establecido en *Perceptual*.

En el ejemplo todos los perfiles de color correspondientes a los espacios CMYK o de color gris no afectan a los elementos gráficos y todos los elementos RGB sin perfil se han asociado al perfil especificado.

Las transformaciones entre espacios de color describen el proceso de obtención de planchas para impresión offset.

$$\begin{pmatrix} x & y & 1 \end{pmatrix} \begin{pmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & 1 \end{pmatrix} = \begin{pmatrix} a_{11}x + a_{21}y + a_{31} \\ a_{12}x + a_{22}y + a_{32} \\ 1 \end{pmatrix}$$

CTM = „a₁₁ a₁₂ a₂₁ a₂₂ a₃₁ a₃₂“

CTM = „1 0 0 1 a b“ corresponde al desplazamiento de (x,y) en (x+a, y+b)

Figura 9.21
Matriz de Transformación
Actual (o en uso) (CTM)

Ejercicio:

Usar los datos de la figura 9.9 para crear una maqueta de imposición. En la figura 9.21 puede ver la definición de la matriz de transformación. Cada punto del espacio bidimensional (x,y) está descrito formalmente en puntos tridimensionales (x,y,1) para permitir las transformaciones, por lo tanto, la matriz de transformación es 3x3. El valor 0,0,1 de la columna derecha no está en la **Matriz de Transformación Actual** (CTM) pero siempre es el mismo, a la izquierda solo quedan 6 valores. Tras multiplicar la matriz, la tercera dimensión se puede eliminar y el resultado puede ser, por ejemplo, CTM= “1 0 0 1 a b” donde podemos observar que los cambios solo se producen sobre los valores (a,b).

10 Impresión

A primera vista resulta paradójico que solo hayan 3 procesos en impresión: *Impresión Digital*, *Barnizado* e *Impresión Convencional* offset, flexo, hueco o serigrafía. Comparándolos, en preimpresión y postprocesado hay muchos más; la diferencia está en el modo de resolver los procesos. En impresión la secuencia suele ser distinta y los procesos independientes (consultar sección 3.3), mientras que en los sistemas de preimpresión y postprocesado no se pueden separar porque están estrechamente relacionados.

En este capítulo solo trataremos los nodos JDF que afectan a la impresión. En el capítulo 9 ya explicamos la comunicación entre dispositivos JDF/JMF (panel de control de la prensa offset) y el Agente o Controlador (el MIS o el servidor de flujo de producción). Expondremos de nuevo un escenario de comunicación para poner en contexto los nuevos nodos: el Agente/Controlador responsable inserta un JDF en una carpeta activa controlada regularmente por el módulo JDF de la imprenta. Alternativamente si el Agente o Controlador envía a la estación de control un comando JDF (*SubmitQueueEntry*) mediante una dirección URL ambas partes pueden utilizarla a modo de servidor de archivos desde donde pueden descargar el archivo JDF con los datos habituales, *Número de Orden (JobID)*, *Nombre de Cliente (CustomerInfo)* y demás recursos para el proceso de impresión. En las siguientes secciones describimos los recursos mencionados.

El Agente/Controlador solicita un *Canal de Comunicación Constante (PersistentChannel)* con el panel de la prensa offset por JDF o mensaje JMF (*Subscription*) para recopilar los datos. La disponibilidad del canal se confirma con una respuesta JMF y las señales JMF solicitadas se transmiten periódicamente al Agente/Controlador por HTTP sin que los implicados necesiten confirmarlas. Esta relación entre panel de control y Agente/Controlador, se explica con la expresión *Envía y Olvida (fire&forget)*. El Agente/Controlador envía un comando (*StopPersistentChannel*) para detener la señal.

El panel de control JDF registra los elementos de auditoría en el *AuditPool* del archivo JDF durante el proceso de impresión. Una vez terminado el trabajo, el sistema de control de la prensa devuelve el archivo JDF modificado mediante carpetas activas o mensajes JMF al Agente o al Controlador, indistintamente.

10.1 Impresión Convencional

Ciertas máquinas de impresión convencional incluyen procesos en línea de preimpresión y funciones de acabado, como las prensas digitales que crean sus propias planchas o las rotativas de bobina que cortan y pliegan sus trabajos, sin embargo el proceso *JDF ConventionalPrinting* no los contempla y se describen mediante procesos combinados.

Un proceso combinado de Impresión Digital incluye los procesos *JDF Impresión Convencional* y *Ajuste de Imagen (ImageSettings)* e Impresión Offset de Bobina debe combinar los procesos *Impresión Convencional* y *Acabado en Línea (WebInlineFinishing)*.

Muchos parámetros de impresión y postprocesado se definen en preimpresión; si los comparamos, el interfaz de prensa es más simple y aparte de actualizar el estado de sus actividades, informar sobre hojas impresas o recibir especificaciones del MIS o preimpresión, no aporta mucho más y aunque dispone de muchos recursos de entrada (opcionales) solo tiene un recurso de salida.

De toda la información procedente del MIS, preimpresión u otros departamentos ¿qué datos, especificaciones o ajustes son importantes para impresión?

Sistemas JDF para Impresión Offset

Fabricante:	Producto:
Akiyama	InkZone
Goss International	Goss Web Center
Heidelberger Druckmaschinen	Prinect Press Manager
Manroland	Printnet PressManager Sheetfed
Komori Lithographic Presses	K-Station
König & Bauer AG (KBA)	JDFLink for Logotronic
Mitsubishi Heavy Industries	IPC Server II
Ryobi Druckmaschinen	MIS Connection Software
Shinora Machinery Company	Shinora CIP4 Center / Station
Sakurai	InkZone Perfect

Independientemente de que la recepción de los datos sea mediante JDF o cualquier otro formato los datos más significativos son:

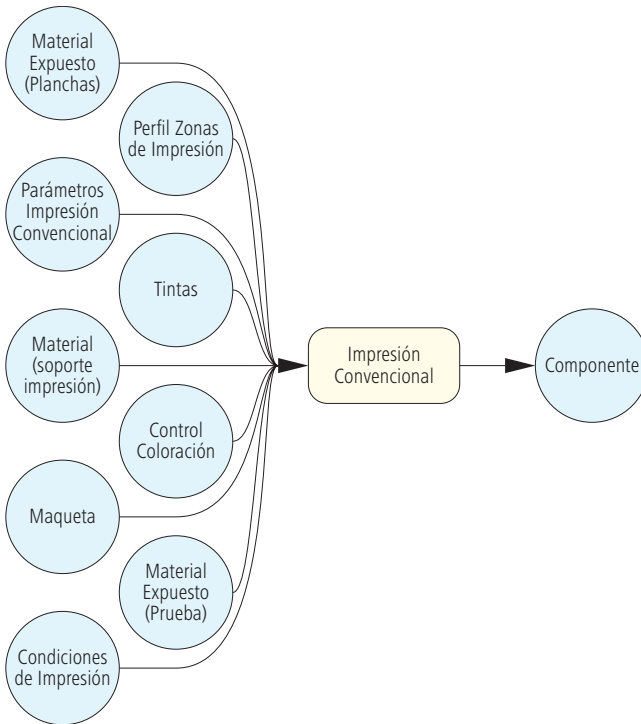
1. Formas de impresión.
2. Tintas.
3. Información sobre sustrato (tamaño, tipo de papel, etc).
4. Datos sobre impresión a doble cara.
5. Datos sobre colores del trabajo.
6. Datos administrativos (número de pedido, cliente, condiciones, tirada o entrega del pedido, etc).
7. Ajuste de tinteros por zonas para impresión offset.
8. Prueba (ferro, contrato, prototipo, vista previa, prueba de pantalla).
9. Posición de los elementos de control sobre el pliego.
10. Condiciones de impresión (valores Lab, densidad de tintas planas, ganancia de punto, objetivos de perfil ICC, etc).

Los seis primeros puntos son necesarios, el siete y el ocho se utilizan a menudo, pero los 2 últimos son imprescindibles. La máquina de imprimir y las indicaciones del proveedor determinan la posición de los elementos de control. Dichos elementos son leídos e interpretados por sistemas de medición en línea o externos si se dispone de ellos. En la actualidad existen pocas máquinas capaces de extraer los metadatos de la forma de impresión para situar los elementos y posicionar el instrumento de medida. Lo más habitual en el área de la impresión offset es que las condiciones de la máquina se controlen directamente a pie de prensa.

Aunque los plazos de entrega se determinan en la orden de pedido, se han incluido en el punto 6 para planificar los turnos y tareas de producción. A continuación, desglosamos el tema en 3 posibilidades:

- La planificación se basa en protocolos del fabricante sin JDF. Estos programas envían las fechas de entrega de los trabajos conectando con el panel de control de máquina.
- Un módulo del MIS planifica el orden de los pedidos. A continuación el programa mencionado en el primer punto supervisado por el MIS, envía un JDF con las fechas, datos técnicos y administrativos al panel de control de la máquina directamente.
- El MIS envía al departamento designado los datos clave para planificar la producción y obtener los suministros necesarios.

Figura 10.1
Impresión Offset
y sus recursos



El primer caso no es relevante para nuestro objetivo. En el segundo, el MIS envía los datos JDF del trabajo directamente al panel de control de la prensa en el orden adecuado. En el tercer caso, el JDF se transmite desde el MIS al programa de planificación de producción. Este archivo JDF tiene diferentes contenidos y destinatarios.

En la figura 3.12 vimos las entradas y salidas de impresión offset.

En la 10.1 las mostramos de nuevo incluyendo los recursos JDF introducidos en los últimos capítulos. Algunos de ellos solo se usan en circunstancias especiales. Por ejemplo, para procesar plantillas preimpresas se necesita un recurso *Component* como entrada. En impresión flexográfica se necesita un recurso extra tipo *Media; MountingTape*, es el adhesivo de doble cara utilizado para montar la camisa en el cilindro. El recurso de entrada *Preview* es necesario para previsualizar el pliego en el panel de la prensa. Ahora veremos los recursos de *ConventionalPrinting*, incluso aquellos que no aparecen en la figura 10.1.

Parámetros y Condiciones de Impresión

JDF distingue muy claramente entre las *Condiciones de Impresión (PrintCondition)* y los *Parámetros de Impresión Convencional (ConventionalPrintingParams)*. *ConventionalPrintingParams* es obligatorio porque alberga los ajustes de la máquina de imprimir. Se puede definir el tipo de secado (UV, Infrarrojo, aire caliente, etc), interrumpir el sistema de mojado para offset seco o elegir la velocidad en pliegos/hora para impresión offset o revoluciones/hora para prensas offset de bobina. También se puede seleccionar el sistema de impresión (offset, flexografía, huecograbado o serigrafía) o el tipo de prensa (offset de pliego o bobina con una o más planchas por cilindro, etc). Esta información se usa para producir el trabajo en una máquina concreta o establecer normas por defecto. En es-

tos estadios los atributos juegan un papel discreto, su utilidad está relacionada con la opción de consultar sobre las competencias del dispositivo asociado, son las denominadas *Funciones de Dispositivo* (*Device-Capabilities*) y las explicaremos al final de la sección 11.2. Este recurso contiene otros parámetros específicos de la prensa, uno de ellos es el tipo de impresión, definido en *WorkStyle*. En la figura 10.2 podemos ver el valor *Voltear* (*WorkAndTurn*-la plancha es la misma para anverso y reverso). También se puede incluir la autorización del cliente o responsable de calidad para enviar el trabajo, el estado de *ConventionalPrinting* es *Detenido* (*Stopped*) hasta ese momento y el detalle mostrado es *Esperando Aprobación* (*WaitingForApproval*).

Figura 10.2
Recurso Parámetros de
Impresión Convencional

```
<ConventionalPrintingParams Class="Parameter" ID="_400"
  PrintingType="SheetFed" Status="Available" WorkStyle="WorkAndTurn" />
```

Los valores especificados en *ConventionalPrintingParams* para ajustar la prensa son muy básicos. En *PrintCondition* se indica la densidad, las curvas de ganancia de punto y los objetivos de color. También se pueden incluir medidas de color espectrofotométricas (temperatura de color, ángulo de medida de 2 o 10 grados, filtros, base de medición, medidas en húmedo o seco, etc).

La realidad es que la industria de impresión offset no ha utilizado este recurso desde la publicación de la norma ISO 12647 [24] y hasta el momento no ha sido necesaria una nueva definición.

La figura 10.3 muestra un ejemplo para la inserción de la curva objetivo de ganancia de punto. Los valores para interpretarla son los mismos de la figura 4.11.

Figura 10.3
Curvas de ganancia de
punto en offset

```
<PrintCondition Name=" Paper type_1_ISO_12647-2" Class="Parameter" ID="_4711"
  PartIDKeys="Side Separation" Status="Available">
  <PrintCondition Side="Front">
    <PrintCondition Separation="Cyan"
      AimCurve="0.0 0.0 0.4 0.56 1.0 1.0" />
    <PrintCondition Separation="Magenta"
      AimCurve="0.0 0.0 0.4 0.56 1.0 1.0" />
    <PrintCondition Separation="Yellow"
      AimCurve="0.0 0.0 0.4 0.56 1.0 1.0" />
    <PrintCondition Separation="Black"
      AimCurve="0.0 0.0 0.4 0.53 1.0 1.0" />
  </PrintCondition>
</PrintCondition>
```

Preajuste de Zonas de Entintado para Impresión Offset

El suministro de tinta en una prensa offset se define mediante zonas individuales denominadas tinteros, estos cubren todo el ancho del cilindro de impresión (entintado corto en sistemas anilox) y su objetivo es ajustar el color a la forma de impresión.

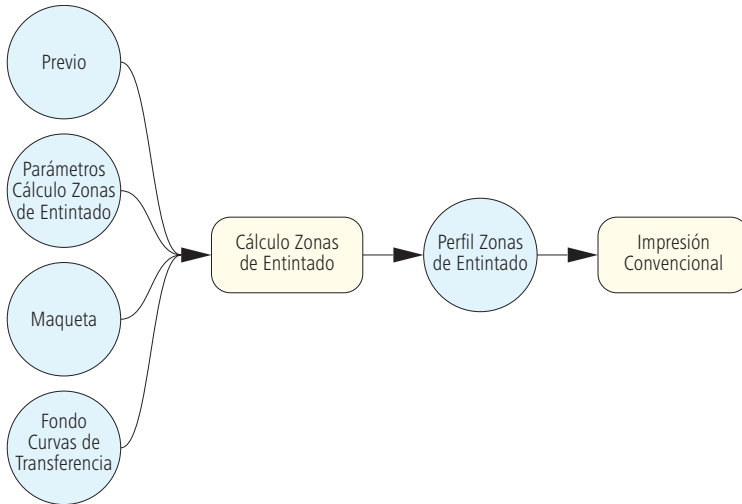


Figura 10.4
Cálculo del preajuste de zonas de entintado en offset

Cada zona dispone de una tecla, rasqueta o regulador para controlar la entrega de tinta, normalmente se usa un sistema electrónico. El preajuste de zonas de entintado forma parte de la creación de la forma de impresión.

Como ya mencionamos en la sección 4.2 los valores para calcular el preajuste de tinteros se obtiene de los datos de preimpresión en formato PPF.

Es tan común que se utiliza como sustituto de JDF y debemos admitir que funciona mejor; si el usuario cambia de formato al menos en este aspecto, no obtendrá ninguna ventaja adicional.

Sin embargo la implementación JDF tiene una enorme ventaja: si enviamos los valores en este formato se pueden empaquetar junto a los demás sin ningún tratamiento especial y lo más importante es que no necesitas pagar una licencia PostScript para interpretar el archivo PPF.

La figura 10.4 muestra un modelo JDF para el cálculo de zonas de entintado. *Previo* (*Preview*) es la previsualización del pliego completo; muestra las zonas por separación de color y la resolución normal son 50 ppi. El tamaño de las zonas se define en *InkZoneCalculationParams* dividiendo la anchura del cilindro impresor proporcionalmente. Las zonas de entintado tienen diferentes anchuras según el tipo de prensa y se suele realizar un cálculo de adaptación. En el ejemplo 10.5 cada zona mide 3,25 cm (92.1259/72x2,54 cm) y son 23 secciones paralelas al eje del cilindro. También es correcta una entrada de zona en el sentido del recorrido del papel y su altura es equivalente al diámetro del cilindro desplegado, pero raramente se utiliza.

La distancia entre las zonas de entintado es exclusiva de cada máquina y la imagen previa simplemente muestra el pliego. Para ajustar las zonas el pliego debe corresponder con la situación de estas, la forma más sencilla es situar el pliego sobre la plancha antes de imprimir, la entrada es el atributo *SurfaceContentsBox* del recurso *Maqueta (Layout)* (tratado anteriormente). Finalmente para alterar los valores tonales de la plancha y de la imagen de vista previa se pueden incluir curvas de transferencia, como vimos en la sección 4.2.

```
<InkZoneCalculationParams Class="Parameter" ID="_25" Status="Available"  
ZoneHeight="1451.338583" ZoneWidth="92.125984" Zones="23" ZonesY="1" />
```

Los valores de preajuste de zonas de entintado se almacenan en el recurso JDF *Perfil Zonas de Tinta (InkZoneProfile)*. Los valores de apertura de tinta no se proporcionan en micrómetros, son datos demasiado científicos. En su lugar se usa un promedio de cobertura por zona con una numeración preestablecida de 0 a 1 que se multiplica por 100 para obtener un valor porcentual. Normalmente se usa una curva de adaptación para determinar las condiciones de impresión propias de la prensa (tinta, papel). El cálculo de apertura por zona no forma parte de la descripción JDF.

Usar los mismos valores de tinta preestablecidos para imprimir una imagen provoca cambios porque cada sustrato absorbe la tinta de diferente manera y la aportación de tinta se debe ajustar a los distintos tipos de papel, ya sean sin recubrimiento (offset), estucados o de arte. Las diferencias se identifican y se almacenan en el panel de control de prensa para reutilizarlas en pedidos de características similares. De este modo, el color del trabajo se ajusta al soporte y las condiciones de impresión de los componentes de control son las correctas antes de haber impreso una sola hoja.

Estos datos no pertenecen a la descripción del pedido y la información no se archiva en el documento JDF pero son valores muy importantes para la producción. Si son erróneos o no se han archivado, los datos de la prensa se deben ajustar para cumplir la especificación de la orden y el ciclo laboral de la imprenta se verá incrementado; se necesitarán más hojas de maculado, mediciones y tiempo con el consiguiente desperdicio en reajustes, repeticiones y copias impresas etc.

Figura 10.5
Parámetros de las
zonas de entintado

En la figura 10.6 vemos un ejemplo del código para describir el *Perfil de Zonas de Entintado (InkZoneProfile)*. El recurso se divide en las separaciones que lo componen y los valores preestablecidos son diferentes por cada color.

Figura 10.6
Valores del preajuste de
zonas de entintado

El ejemplo muestra los valores del color negro. Los 23 valores del atributo *ZoneSettingX* comienzan por la izquierda y contienen los valores de cobertura por zona limitados a 2 decimales: 0,49%, 0,86%, 4,09%, 5,25%, etc. El valor promedio de cobertura del conjunto es 4,12% y se expresa en el atributo *ZoneSettingY*.

```
<InkZoneProfile Class="Parameter" ID="_3" PartIDKeys="SignatureName SheetName
Side Separation"...>
  <InkZoneProfile SignatureName="S1">
    <InkZoneProfile SheetName="Cover">
      <InkZoneProfile Side="Front">
        <InkZoneProfile Separation="Black" Status="Available"
          ZoneWidth="92.125984" ZoneSettingsX="0.004947916666669755
          0.008625565610862806 0.040955175339369494
          0.05258389894419598 0.036772011689294073
          0.04679616798642827 0.03305889423077225
          0.04669306184012362 0.023023897058826546
          0.03196920955882654 0.026367305335598803
          0.022763303638766233 0.01726845305430168
          0.027979296285825045 0.03681620003771036
          0.0802906414969863 0.07557833710407524
          0.07194204845399976 0.05891538225867557
          0.08456872171945984 0.08174302413273286
          0.029526241987182487 0.009106099170440482"
          ZoneHeight="1451.338583"
          ZoneSettingsY="0.04123003711309234" />
        <InkZoneProfile ... />
      </InkZoneProfile>
    </InkZoneProfile>
  </InkZoneProfile>
</InkZoneProfile>
```

Marcas de Control

En las formas de impresión se incluyen habitualmente varias marcas de control. Estas se pueden clasificar según su función:

- Elementos de control para producción de formas impresas.
- Elementos de control y supervisión de impresión.
- Elementos de control y supervisión de postimpresión.

Los elementos de control más relevantes que conciernen a la impresión son:

- Registro o, concretamente, marcas de registro.
- Tiras de control de impresión.
- Marcas de páginas o equipamiento, denominadas tira/retira.
- Signatura o diseño de las formas de impresión.

Como es lógico, debemos situar las marcas sobre el pliego: las de corte se colocan en los límites, las tiras de control en medio y las que indican los límites inferior y superior paralelas al eje del cilindro. Si el operario utiliza un dispositivo manual para comprobar las medidas la posición carece de importancia. Cuando la medición se realiza con sistemas automáticos en línea, se sitúan en lugares concretos para que los sensores se puedan autoajustar.

Existen otros sistemas para controlar el color en línea, unos miden la densidad y otros son específicos para offset, flexografía o huecograbado, incluido uno que realiza mediciones espectrofotométricas en línea del pliego completo.

Se usan cámaras CCD para leer las marcas de registro y controlar la desviación lateral, diagonal y circunferencial. Los elementos de control son situados en la maqueta de imposición del programa de montaje por un operador de preimpresión. También sabemos que el software puede calcular la ubicación por si mismo mediante las denominadas “marcas dinámicas” (ver sección 9.2).

Por lo tanto, la posición de las marcas es un ejemplo de transmisión de datos entre departamentos y es una tarea que puede asumir JDF. En la figura 10.7 mostramos un recurso *Layout* que incorpora varios recursos *MarkObject*.

Formas de Impresión y Sustratos

Las formas de impresión pueden ser planchas, cilindros de huecograbado, camisas flexográficas o pantallas de serigrafía y solo pueden estar en dos estados: en blanco o con datos de imagen. En impresión offset se utiliza el término fotográfico “expuestas” o “no expuestas” para hablar de las planchas y se puede usar perfectamente porque todas las planchas son sensibles a la luz. El revelado de las planchas se añade al proceso *Ajuste de Planchas* (*PlateSettings*).

En JDF las planchas sin exponer se describen como recursos materiales *Media* donde el atributo *Media Type* es la propia plancha. El sustrato o tipo de material se define con el mismo recurso y puede ser Papel (*Paper*), Aluminio (*Foil*) o Cartón Ondulado (*CorrugatedBoard*).

```
<Layout Class="Parameter" DescriptiveName="Lay" ID="_111" Name="Lay"
PartIDKeys="SignatureName SheetName Side" Status="Available">
  <Layout Name="SIG1" SignatureName="SIG1">
    <Layout DescriptiveName="Sheet" Name="Sheet" SheetName="Sheet"
SourceWorkStyle="WorkAndTumble" Status="Available"
SurfaceContentsBox="0 0 2111.81102362 1714.96062992">
      <Media Class="Consumable" Dimension="2111.81102362 1714.96062992"
ID="_3062" MediaType="Plate" Status="Available" />
      <Layout DescriptiveName="Recto" Side="Front"
Status="Available"
SurfaceContentsBox="0 0 2111.81102362 1714.96062992">
        ...
        <MarkObject CTM="1 0 0 1 0 0"
ClipBox="0 0 2111.81102362 1714.96062992" Ord="0">
          <RegisterMark Center="2028.42519684 659.11417322"
Class="Parameter" MarkType="8AR_R_106x164" Rotation="0" />
          <RegisterMark Center="83.38582677 659.11417322"
Class="Parameter" MarkType="8AR_L_106x164" Rotation="0" />
          <ColorControlStrip Center="1059.44881889 809.11811023"
Class="Parameter" Rotation="0" Size="1798 28"
StripType="FOGRA_5_F74_740x10" />
        </MarkObject>
      </Layout>
    </Layout>
  </Layout>
```

Figura 10.7
Recurso Maqueta (Layout)
con objetos Marcas (Mark)

El proceso de impresión necesita planchas expuestas sin contar las planchas ciegas o blancas utilizadas ocasionalmente en la impresión rotativa de periódicos.

Las planchas expuestas se enumeran en el recurso *ExposedMedia*. En la práctica, las imprentas utilizan un solo tipo de plancha y cada prensa utiliza su propio tamaño. Existe un proceso que indica al impresor cuando tiene las planchas preparadas para imprimir, este recurso retransmite una orden automáticamente a la prensa cuando ya están disponibles todas las formas o planchas.

Esta información es lo que muestra la figura 10.8 y además contiene una referencia al recurso *Media* con el tamaño y tipo de plancha.

En el código JDF de la figura 10.9, podemos ver en la partición del recurso *Media* el sustrato para la cubierta del libro.

- *Tamaño* de hoja 61/43 cm. (*Dimension*).
- Papel *Clase 1* norma ISO12647/2:2004 (*Grade*).
- *Dirección del grano* (*GrainDirection*).
- *Grosor* de papel 215 micrómetros (*Thickness*).
- *Peso* del papel de 215g/m2 (*Weight*).
- *Recubrimiento del anverso brillante* (*FrontCoatings*).
- *Recubrimiento del reverso mate* (*BackCoatings*).
- *Marca* Chromolux (*Brand*).

Figura 10.8
Datos de las planchas para
el proceso de impresión

```
<ExposedMedia Class="Handling" ID="_123" PartIDKeys="SignatureName SheetName
Side Separation"...>
  <ExposedMedia SignatureName="Signatur_1">
    <ExposedMedia SheetName="Cover" Status="Available">
      <ExposedMedia Side="Front" Status="Available">
        <ExposedMedia Separation="Cyan" Status="Available" />
        <ExposedMedia Separation="Magenta" Status="Available" />
        <ExposedMedia Separation="Yellow" Status="Available" />
        <ExposedMedia Separation="Black" Status="Available" />
      </ExposedMedia>
      <ExposedMedia Side="Back" Status="Available">
        <ExposedMedia Separation="Cyan" Status="Available" />
        ...
      </ExposedMedia>
    <MediaRef rRef="_1234">
      <Part SheetName="Cover" SignatureName="Signatur_1" />
    </MediaRef>
  </ExposedMedia>
</ExposedMedia>

<Media Brand="745x605_Azura" Class="Consumable" Dimension="2111.811
1714.961" ID="_1234" MediaType="Plate" PartIDKeys="SignatureName SheetName"
Status="Available">
  <Media Class="Consumable" Dimension="2111.81102362 1714.96062992"
SignatureName="Signatur_1" Status="Available">
    <Media Brand="745x605_Azura" Class="Consumable"
Dimension="2111.811 1714.961" MediaType="Plate" SheetName="Cover"
Status="Available" />
  </Media>
</Media>
```

También pueden incluirse atributos adicionales como el grado de opacidad o el valor Lab del papel. Estos datos podrían ser proporcionados en las listas de precios del proveedor de papel, por ejemplo, pero por desgracia esto no es lo habitual.

```

<Media Class="Consumable" ID="_111" MediaType="Paper"
PartIDKeys="SignatureName SheetName" Status="Available"...>
  <Media Class="Consumable" SignatureName="Signature_1" Status="Available">
    <Media SheetName="Cover" Class="Consumable" ProductID="_999"
      Status="Available" Dimension="1729.1338582677165 1218.8976377952756"
      Grade="1" GrainDirection="LongEdge" Thickness="215.0" Weight="215.0"
      FrontCoatings="Glossy" BackCoatings="Matte" Brand="Chromolux" />
  </Media>
</Media>

```

Figura 10.9
Información del sustrato

Colores y Métodos de Control del Color

El término “color” se utiliza para definir muchos conceptos como son: la secuencia de colores en prensa, los diferentes colores posibles de las dos caras de una hoja, las propiedades colorimétricas de la tinta, nombres de colores en los contenidos, el modelo de color de salida del dispositivo, etcétera. Esta información se almacena en diferentes recursos del JDF.

El recurso *Ink* contiene la lista de tintas utilizadas en impresión. En su versión más simple son únicamente los colores de proceso, como vimos en la figura 8.15, la cara frontal es CMYK y el reverso negro. Obviamente, también se pueden incluir barnices o tintas especiales. Sin embargo, el recurso *ColorPool* donde se enumeran todos los colores de una orden o pedido, necesita que los nombres de las separaciones y de las tintas se correspondan incluidos los no utilizados en prensa por cualquier causa.

También se incluyen datos como *Tipo de Color* (*ColorType*) que detalla si la tinta es la normal o translúcida, opaca o transparente y la *Densidad Neutra* de los colores (*NeutralDensity*).

En el capítulo 9 Preimpresión comprobamos la importancia de las propiedades del color para el proceso de *Trapping* o *Reventado*. En este apartado se pueden incluir valores ColorLab y fórmulas para describir colores CMYK y tintas especiales.

La figura 10.10 nos muestra un ejemplo del recurso *ColorPool*. Cada color *CMYK* expresa el valor tonal con un porcentaje, pero como es habitual, es normalizado a 1 en vez de a 100. *ColorType* es una excepción: es *Normal*, por lo tanto translúcido. La excepción de color se utiliza para perfilar la hoja de prueba.

Las normas de *Trapping* con respecto a los demás no se aplican a este color y no será impreso, por lo tanto no se expondrá ninguna plancha.

```

<ColorPool Class="Parameter" ID="_500" Status="Available">
  <Color CMYK="1.0 0.0 0.0 0.0" ColorType="Normal" Name="Cyan"
    NeutralDensity="0.61" />
  <Color CMYK="0.0 0.0 0.0 1.0" ColorType="Normal" Name="Black"
    NeutralDensity="1.7" />
  <Color CMYK="0.0 1.0 0.0 0.0" ColorType="Normal" Name="Magenta"
    NeutralDensity="0.76" />
  <Color CMYK="0.0 0.0 1.0 0.0" ColorType="Normal" Name="Yellow"
    NeutralDensity="0.16" />
  <Color CMYK="1.0 0.0 1.0 0.0" ColorType="DieLine" Name="ProofColor" />
  <Color CMYK="0.2 0.3 0.4 0.5" ColorType="Normal" Name="PANTONEDeepBlue"
    Lab="20. 30. 40.">
</ColorPool>

```

Figura 10.10
Recurso ColorPool

Vamos a enumerar una vez más el contenido del último recurso relacionado con la descripción y definición del color *ColorantControl* (secciones 6.4, 8.2 y 9.1 y figuras 6.18, 8.12, 9.8 y 9.13). Este recurso alberga toda la información necesaria para asignar los colores de los datos de contenido a los de salida de dispositivo.

Dicha información es irrelevante para la prensa de imprimir, sin embargo el subelemento *DeviceColorantOrder* especifica la secuencia de los colores en prensa necesaria para el proceso que generalmente es negro, cian, magenta y amarillo (figura 10.11).

Figura 10.11
Recurso ColorantControl

```

<ColorantControl Class="Parameter" ID="_2201" PartIDKeys="SignatureName
SheetName Side" ProcessColorModel="DeviceCMYK" Status="Available">
  <ColorPoolRef rRef="_500" />
  <DeviceColorantOrder>
    <SeparationSpec Name="Black" />
    <SeparationSpec Name="Cyan" />
    <SeparationSpec Name="Magenta" />
    <SeparationSpec Name="Yellow" />
    <SeparationSpec Name="PANTONEDeepBlue" />
  </DeviceColorantOrder>
  ...
</ColorantControl>

```

Prueba y Previsualización

Por regla general, el impresor recibe una prueba de contenido (prueba ferro) reproducida sin gestión del color por una impresora de inyección de tinta. Además, si está configurado en red y seleccionamos un trabajo en el panel de control nos muestra una miniatura del pliego.

En el caso de la imprenta Y, tratado en la sección 2.2, la estación de control muestra la prueba en una pantalla calibrada, el entorno de trabajo está equipado con iluminación normalizada y los colo-

res son calibrados correctamente para mostrar pruebas de contrato y cumplir con la normativa de calidad. Como viene siendo habitual, las imágenes de vista previa y las pruebas se describen utilizando recursos JDF.

Para mostrar la vista previa o prueba en pantalla se utiliza un elemento *RunList* y para la prueba impresa un elemento *Component*, como ya expusimos en la sección 9.5.

Especialmente en el sector del embalaje, encontramos casos donde las pruebas de color se reproducen con una filmadora o dispositivos capaces de reproducir mapas de bits rasterizados denominados pruebas tramadas; se representan mediante una variación especial del recurso *Material Expuesto (ExposedMedia)*.

Componente

Los pliegos impresos son productos intermedios de un proceso global y salida del proceso de impresión. En lenguaje JDF significa que la salida del proceso *Impresión Convencional (ConventionalPrinting)* es un *Tipo de Componente (ComponentType) Hoja (Sheet)*, sin embargo estos pueden dividirse antes y después de se impresos. Aclaramos esta distinción a continuación.

```
<Component Class="Quantity" ComponentType="Sheet" ID="_123"
  PartIDKeys="SignatureName SheetName Condition" Status="Unavailable">
  <Component SignatureName="Signature_1">
    <Component SheetName="Cover" />
  </Component>
</Component>

<ComponentLink Amount="1" Usage="Output" rRef="_012">
  <AmountPool>
    <PartAmount Amount="2040.0">
      <Part Condition="Good" SheetName=" Cover "
        SignatureName=" Signature_1" />
    </PartAmount>
    <PartAmount Amount="364.0">
      <Part Condition="Waste" SheetName=" Cover "
        SignatureName=" Signature_1" />
    </PartAmount>
  </AmountPool>
  <Part SheetName=" Cover " SignatureName=" Signature_1" />
</ComponentLink>
```

Figura 10.12
Componentes del
pliego de impresión

El recurso *Enlace de Componentes (ComponentLink)* de la figura 10.12, distingue en *Condición (Condition)* entre hojas buenas (*Good*) y residuales (*Waste*).

En *Fondo de Recuento (AmountPool)* se indica el número de pliegos correctos (1040) y maculaturas (364). En cierto sentido es una especie de “bloc de notas” donde el contador anota el recuento de los recursos generados y utilizados durante el proceso.

Las implementaciones JDF más habituales son en imprentas offset de hoja. Las tasas de integración son muy bajas en impresión offset de bobina, huecograbado, flexografía o serigrafía y suele estar limitada a las áreas de preimpresión y software MIS (ver capítulo 12).

Actualmente desconocemos instalaciones JDF en huecograbado, flexografía o serigrafía aunque estos procesos de impresión están contemplados en el proceso *ConventionalPrinting*. Los fabricantes de las máquinas correspondientes han mostrado poco interés por el tema hasta la fecha. Muchos de ellos ni siquiera son miembros de la organización CIP4 y sus razones podrían ser las siguientes:

- Los promotores iniciales de JDF fueron fabricantes de prensas offset o la demanda en las demás tecnologías de impresión ha sido menor y los proveedores no lo han considerado antes.
- La integración de preimpresión con las tecnologías de impresión flexográfica y huecograbado se realiza en distintas empresas y es complicada.
- La transferencia de valores preestablecidos para el cálculo de zonas de entintado en flexo y hueco es ineficaz y los costes son excesivos, a diferencia de la tecnología offset.
- Offset dispone de normas estandarizadas, pero estas tecnologías suelen usar ajustes específicos de empresa/cliente.
- La gran variedad de sustratos complica la automatización.
- Con respecto a los volúmenes de impresión, la puesta a punto en offset es más barata que en hueco/flexo y se debe contabilizar.
- Los sistemas de control de órdenes suelen ser a medida para un reducido grupo de grandes corporaciones.

Sin embargo, seguramente esta situación cambiará en el futuro. La mensajería JDF interesa a todo el mundo porque optimiza los consumos de material, informa del estado de máquinas y pedidos y racionaliza la producción.

10.2 Impresión Digital

La tecnología de impresión digital es conocida por sus puntos fuertes y beneficios especialmente significativos en la reducción de los tiempos de producción, las tiradas cortas y la personalización.

Por lo tanto, la tecnología JDF es un formato potencialmente relevante para la impresión digital y un requisito previo para automatizar completamente el proceso.

La impresión digital se divide en 2 grandes áreas de aplicación con fronteras difusas:

- Impresión Digital en Oficinas y Copisterías (Imprenta Rápida).
- Sistema Profesional de Impresión Digital.

Las opciones de imposición, gestión de color y acabado en línea distinguen a los sistemas profesionales de las copisterías. Sus respectivas especificaciones ICS las describen: *Impresión Digital de Oficina (Office Digital Printing ICS)* e *Impresión Digital Integrada (Integrated Digital Printing-IDF ICS)*. Los sistemas digitales integrados en redes JDF son parte de *Flujos de Trabajo Híbridos* donde todos los datos se preparan en el WMS tanto para impresión digital como para offset. La compatibilidad con JDF automatiza la producción de planchas y contempla el grabado de cilindros para hueco. Existen casos donde los datos JDF del MIS son captados por el sistema de huecograbado y este devuelve informes al MIS.

Tabla 10.13

No.	Sistema Integrado de Impresión	Impresión Digital en Oficina/Tienda de Copias
1	Preparación de Maqueta (LayoutPreparation)	Preparación de Maqueta (LayoutPreparation)
2	Imposición? (Imposition?)	Imposición (Imposition)
3	¿Conversión de Espacio de Color? (ColorSpaceConversion?)	
4	Interpretación (Interpreting)	Interpretación (Interpreting)
5	¿Conversión de Espacio de Color? (ColorSpaceConversion?)	
6	Rasterizado (Rendering)	Rasterizado (Rendering)
7	¿Conversión de Espacio de Color? (ColorSpaceConversion?)	
8	¿Tramado? (Screening?)	
9	¿Imposición? (Imposition?)	
10	Impresión Digital (DigitalPrinting)	¿Impresión Digital? (DigitalPrinting?)
11	¿Plegado? (Folding?) ¿Grapado? (Stitching?),	¿Plegado? (Folding?)
12	¿Guillotinado? (¿Trimming?), ¿Agujereado? (HoleMaking?), ¿Cubierta? (CoverApplication?), ¿Encuadernado? (SpineTaping?)	¿Grapado? (Stitching?)

Las diferencias técnicas entre impresión digital y offset de hoja son:

- El proceso JDF de filmación (*ImageSetting*) en Impresión digital es innecesario porque no usa elementos físicos.
- En impresión digital el acabado en línea está integrado en la máquina.

En este último no necesitamos describir los pasos intermedios, es un proceso JDF combinado, La tabla 10.13 muestra los procesos posibles ordenados según las especificaciones ICS; *Sistema Integrado de Impresión* muestra los procesos *Imposition* y *ColorSpaceConversion* varias veces y el interrogante nos indica que son opcionales. Esto significa que una vez definido su lugar, ambos deben estar siempre en la misma posición dentro del flujo de trabajo a diferencia de los procesos de acabado donde la secuencia puede ser arbitraria y todos los procesos son opcionales.

Impresión Digital en Oficina o Copistería no suelen gestionar el color ni disponen de opciones profesionales de acabado.

Debido a la gran variedad de procesos implicados, este proceso combinado puede tener muchos recursos de entrada. Siempre se obtiene un recurso *Componente* como salida del proceso combinado. Los resultados intermedios de los procesos son recursos tipo *Component* o *RunList*. La figura 10.14 muestra un ejemplo, sin todas las opciones posibles, basado en nuestra propia experiencia. La figura 10.15 es el código correspondiente.

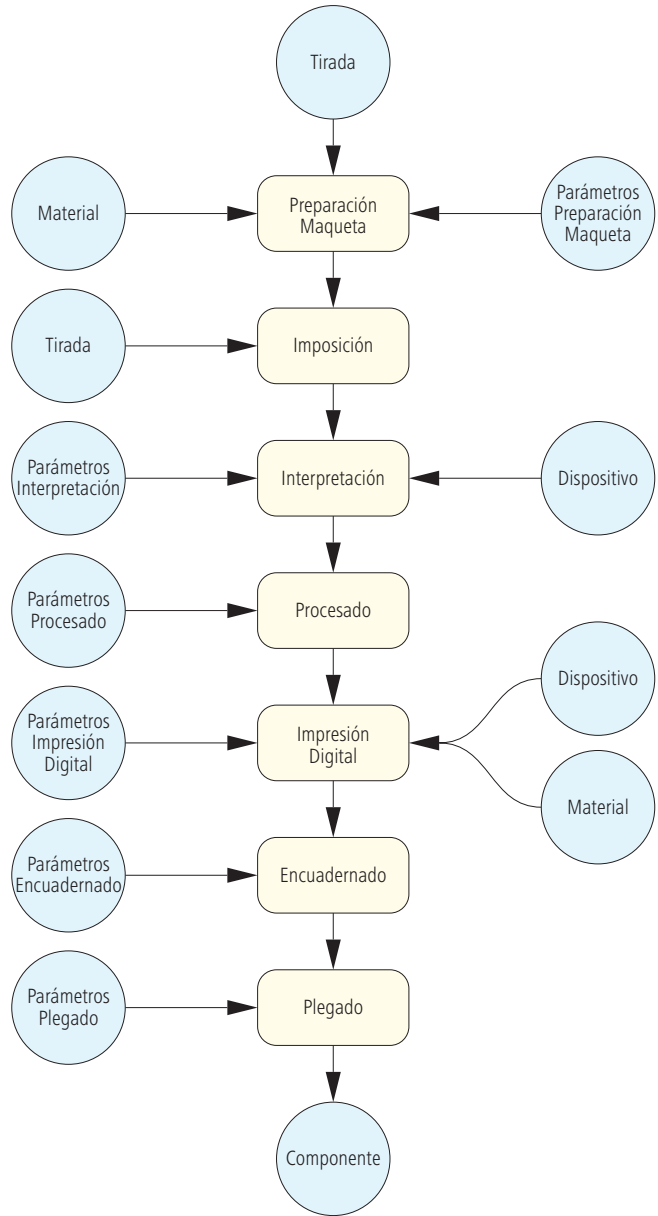


Figura 10.14
Ejemplo práctico de impresión digital

```

<?xml version="1.0" encoding="UTF-8"?>
<JDF Status="Ready" Type="Combined" Types="LayoutPreparation Imposition
Interpreting Rendering DigitalPrinting Stitching Folding" Version="1.3" ...>
...
  <ResourceLinkPool>
    <ComponentLink Amount="65" CombinedProcessIndex="5" Usage="Output"
      rRef="_2503">
      <Part SignatureName="Sig1" />
      <Part SignatureName="Sig2" />
      <Part SignatureName="Sig3" />
    </ComponentLink>
    <LayoutPreparationParamsLink CombinedProcessIndex="0" Usage="Input"
      rRef="_2504" />
    <MediaLink CombinedProcessIndex="0 4" Usage="Input" rRef="_0918" />
    <DigitalPrintingParamsLink CombinedProcessIndex="4" Usage="Input"
      rRef="_2507" />
    <DeviceLink CombinedProcessIndex="2 4" Usage="Input" rRef="_2509" />
    <RenderingParamsLink CombinedProcessIndex="3" Usage="Input"
      rRef="_2511" />
    <InterpretingParamsLink CombinedProcessIndex="2" Usage="Input"
      rRef="_2510" />
    <StitchingParamsLink CombinedProcessIndex="5" Usage="Input"
      rRef="_2513" />
    <StitchingParamsLink CombinedProcessIndex="6" Usage="Input"
      rRef="_2514" />
    <RunListLink CombinedProcessIndex="0 1" ProcessUsage="Document"
      Usage="Input" rRef="_9437" />
  </ResourceLinkPool>
</JDF>

```

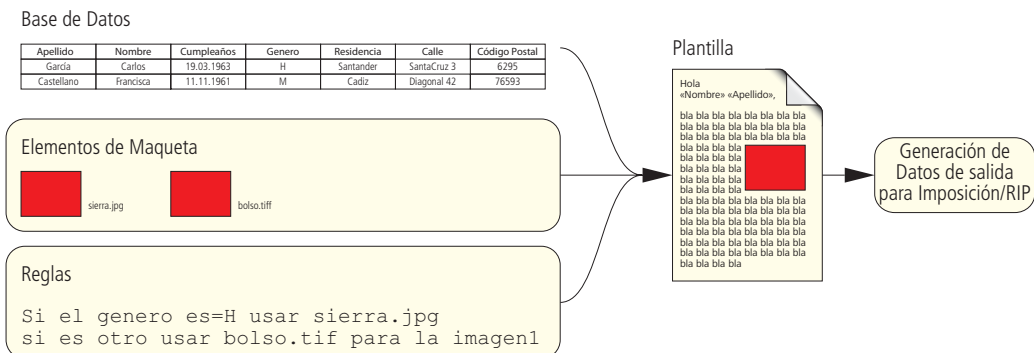
Figura 10.15
Ejemplo práctico de
impresión digital

Preparación de Maqueta (LayoutPreparation) es similar al proceso tratado en la sección 9.2 *Montaje (Stripping)*. Ambos deben producir un diseño de imposición como recurso de salida. La información que necesitan para generar el diseño proviene de sus recursos de entrada. *StrippingParams* es el recurso de entrada del proceso *Stripping* y el de *Generación de Maqueta (LayoutGeneration)* es *LayoutGenerationParams*. La diferencia entre ambos radica en su entorno de aplicación, *LayoutPreparation* se usa en impresión digital y *Stripping* en preimpresión offset coordinado con el MIS.

Existe la posibilidad de que en futuras versiones de la especificación JDF se produzcan cambios y los recursos *LayoutPreparationParams* que figuran actualmente en la versión 1.4 sean desestimados.

Personalización

La impresión personalizada o dato variable (VDP) es un sistema para adaptar cada cara, unidad u hoja impresa a las preferencias del cliente cambiando elementos del diseño como textos, gráficos o imágenes. En Variable Data Printing (VDP) una parte del diseño es constante en todas las hojas y otra es sustituida en cada copia generada por un archivo de control o base de datos.



Habitualmente el pedido se compone de las siguientes subtarefas (figura 10.16):

- Preparación de base/tablas de datos de destinatarios.
- Diseño de plantilla por cara/unidad con elementos estáticos y variables realizado con software (plugings) incorporado a los programas de maquetación.
- Normalización del sistema para la sustitución de las variables de la plantilla con la base de datos.
- Fusión de los datos para enviarlos al RIP de impresión digital.

Se pueden usar PDF normales para transferir los datos personalizados al RIP. A continuación, se procesa cada cara de la hoja individualmente incluyendo los elementos estáticos. La parte variable está integrada entre los elementos estáticos del diseño, pero el método es poco efectivo. En su lugar se suele utilizar un lenguaje específico que transmite los elementos estáticos y variables simultáneamente al RIP con mejores resultados.

Figura 10.16
Tareas subyacentes para personalizar la impresión

Las principales herramientas disponibles en la actualidad son: **PPML Impresión Personalizada basada en Lenguaje Marcado XML** [11] [36] [37] [38] y el formato patentado específico para Artes Gráficas **PPML/GA** utilizado por numerosos fabricantes de soluciones.

PPML/VDX (Variable Data Exchange) es otra opción basada en PDF, PPML y la nueva solución **PDF/VT (Variable Transactional)** propiedad de Adobe Systems, se ha diseñado especialmente para el intercambio de datos variables en trabajos etiquetados como JDF. Según algunos expertos PDF/VT [25] reemplazará a PPML por su integración en la tecnología **PrintEngine 2 OEM RIP** [1] original de Adobe Systems. Existen otros formatos patentados por proveedores como el mencionado PPML/GA que no trataremos en el libro.

PPML se presentó a la vez que JDF en el año 2000. Es un lenguaje independiente de proveedor presentado por la asociación sin ánimo de lucro Iniciativa de Impresión Digital PODi fundada en 1996.

Al hablar de impresión digital, su análisis nos plantea la siguiente pregunta: ¿PPML compite o complementa JDF?. Para responderla examinaremos los aspectos más relevantes del formato.

PPML permite integrar objetos digitales como textos, imágenes y páginas en diferentes formatos de datos (EPS, PDF, JPEG, TIFF, etc). Los objetos se incorporan al archivo PPML mediante referencias y se pueden mover y transformar. Una vez los ha interpretado el RIP se pueden almacenar en la memoria caché para su reutilización pero conservan el formato original, no se incrustan en PPML. Dicho de otro modo, las páginas PPML solo describen la construcción de las páginas combinando los objetos.

JDF siempre inicia el flujo de trabajo con páginas completas y tanto PDF como PPML describen páginas pero la estructura de este último es especial para datos variables. Por lo tanto JDF y PPML son complementarios, aunque ciertas razones lo contradicen:

- El proceso *LayoutElementProduction* de JDF también describe la creación de página.
- PPML también incluye metadatos opcionales para el flujo de trabajo, como la descripción del diseño de imposición [37].

En ocasiones ambos formatos se solapan pero se complementan en muchas otras. En PPML, por ejemplo, no se pueden concretar las propiedades del sustrato, sin embargo en el recurso JDF *Media* se definen perfectamente.

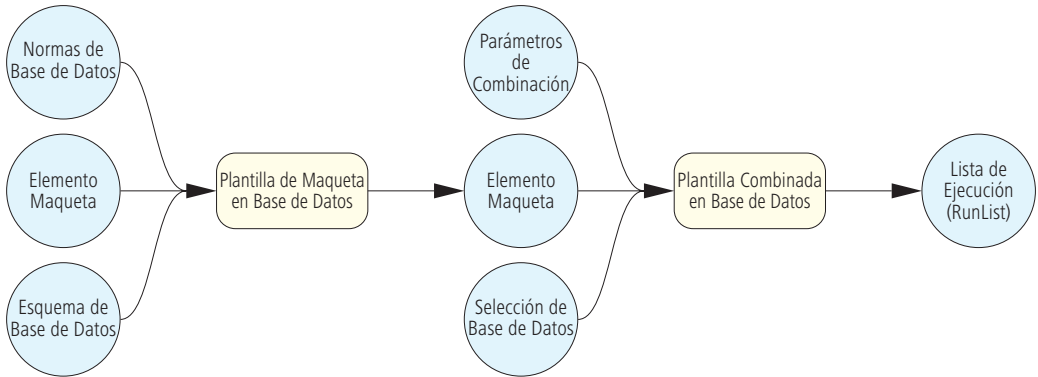


Figura 10.17
Modelo JDF de
impresión personalizada

De hecho los archivos PPML pueden contener referencias a archivos/datos JDF pero en la versión 2.2 se eliminó la posibilidad y en el futuro PPML solo servirá para describir el contenido. Por el contrario, un recurso *RunList* JDF puede hacer referencia a datos PPML (especificación de las normas entre ambos formatos [39]). El atributo *IgnorePDLImposition* del recurso JDF *LayoutElement*, por ejemplo, indica si los datos para maquetar proceden de PPML o JDF.

La figura 10.17 muestra el modelo JDF de datos variables de nuevo. El recurso *Lista de Ejecución (RunList)* representa un PPML (o archivo en cualquier otro idioma VDP). A la izquierda vemos la entrada que define la situación de los elementos estáticos y variables en el documento mediante enlaces con la base de datos. La *Plantilla de Maqueta en Base de Datos (DBDocTemplateLayout)* es la típica entrada de un proceso *Imposition*.

Para incorporar los elementos en la plantilla los programas deben cumplir unas normas pero algunos no las consideran relevantes y las incluyen como comentarios en elementos *NormasDB (DBRules)*.

El tipo de base de datos puede ser SQL, XML o texto separado por comas y se especifica en *EsquemaDB (DBSchema)*, también puede incorporar comentarios referentes a la base de datos. Los contenidos y enlaces estáticos/variables URL se incluyen en *Elementos de Maqueta (LayoutElement)*. El proceso *Plantilla de Maqueta DB (DBDocTemplateLayout)* genera un nuevo recurso *Maqueta (LayoutElement)*. El valor del atributo *Plantilla (Template)* debe ser *Cierto (True)* para ser reconocido como tal.

Finalmente, el proceso *Plantilla Combinada DB (DBTemplateMerge)* genera un archivo en un lenguaje similar a PPML que informa de la situación del documento en el sistema para que *Parámetros de Combinación (DBMergeParams)* pueda encontrarlo.

El recurso *Selección DB* (*DBSelection*) conserva la siguiente información escrita en el mismo lenguaje de programación (por ejemplo SQL) que la base de datos: índices de registro, dirección URL y secuencias de acceso a la base de datos.

11 Postimpresión

Antes de ser publicada la especificación JDF algunas máquinas de acabado y equipos de postimpresión ya funcionaban en red. Un ejemplo que ya tratamos en el capítulo 4.2 son las guillotinas de alta velocidad con captura de datos PPF o las soluciones basadas en la lectura de códigos de barras impresos patentadas por sus desarrolladores. El sistema consistía en imprimir un código de barras en el parte de trabajo con los datos más relevantes para el plegado del trabajo como son el tamaño de hoja y la posición de los hendididos, posteriormente era registrado con un lector de códigos EAN.

Existe otra solución todavía en uso que se explica mejor con un ejemplo: cuando se inicia un trabajo en un programa de corte, se definen respectivamente las tareas de la guillotina y la plegadora. Los costes de preparación de estas máquinas son altísimos y para evitar paradas y no afectar a la productividad, los ordenadores se configuran antes de realizar las instalaciones pertinentes por red, reduciendo los tiempos muertos del equipo de acabado. Además, estos sistemas están programados para recoger datos de pedido y distribuirlos a los equipos de producción, especialmente de plegado y grapado. Estos son tiempos de preparación, estado de las máquinas y tasas de producción. Los fabricantes han desarrollado soluciones que integran las máquinas de impresión, los paneles de control y los protocolos específicos entre el programa de planificación y los equipos. Pero estas configuraciones en red no son muy abundantes y todavía son más escasas las conexiones entre JDF y acabado.

Existen dos áreas potenciales de acabado donde incluir JDF/JMF:

- El terminal de la consola de la máquina.
- Los sistemas de preparación de trabajos de las máquinas de acabado.

Volviendo a la terminología CIP4, el dispositivo del primer caso se comunica directamente por red local con la máquina o máquinas de producción. En el segundo punto, el dispositivo JDF comunica directamente por red local con la estación de trabajo utilizada. Los valores JDF importados se utilizan en ambos casos para realizar el ajuste predeterminado, pero el usuario puede modificarlo en el software posteriormente. En el segundo caso, los programas de producción de acabado pueden comunicar por red con las máquinas en el formato propietario.

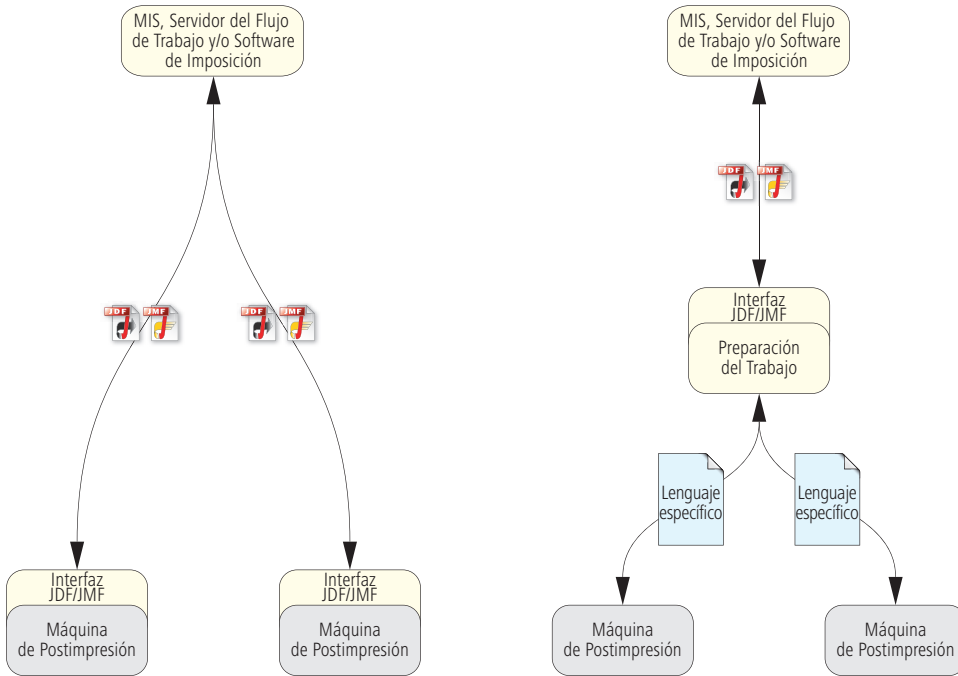


Figura 11.1
Diagramas de flujo de las
dos configuraciones
JDF/JMF más
importantes para
postimpresión

La figura 11.1 muestra los diagramas de flujo de ambos modelos, donde podemos ver las diferencias entre las dos configuraciones. Si fuera necesario, la estación de trabajo de preparación tiene la capacidad de organizar y distribuir los pedidos a diferentes máquinas, liberando de la tarea al operador de máquina(s). En cambio si el interfaz JDF/JMF está incorporado en la máquina, el MIS y el operador pueden especificar la secuencia de producción y reducir al mínimo los tiempos de cambio entre trabajos. La combinación de los interfaces JDF/JMF con las máquinas de producción puede plantear ciertas dificultades al actualizar el software.

Ejemplos de conectividad JDF compatible con Postimpresión (2009)

Manufacturer:

Ferag AG
Heidelberger Druckmaschinen
Hohner Maschinenbau GmbH
Horizon International Inc.
MBO
Müller Martini

Main Product:

iQ
Princt Postpress Manager
ixFrame (der ixact GmbH)
i2i
Datamanager
Connex

Por otra parte, la solución de preparación de trabajos solo tiene sentido si vamos a distribuir los trabajos entre varias máquinas de producción. Los dispositivos JDF de acabado pueden recibir la información JDF/JMF desde un MIS, el servidor del flujo de trabajo en producción o directamente desde el programa de montaje. Los dos últimos son los que realmente definen las posiciones de corte y plegado, siempre y cuando sea posible transmitir la información. También es factible una configuración donde el programa de montaje devuelve información al MIS o al servidor del flujo de trabajo que da servicio al sistema de acabado a su vez. Otra posibilidad es que el programa de montaje controlado completamente por el MIS y sin capacidad para alterar la posición de ninguna página realice el diseño de imposición y determine directamente la posición de las marcas de corte y plegado para el sistema de acabado.

11.1 Corte por Guillotina

Los pliegos en bruto son productos intermedios que deben ser cortados; este proceso se suele realizar con guillotinas o cortadoras de alta velocidad. El guillotinado de pliegos antes de imprimir sirve para dar el tamaño deseado a la hoja y cortarla a escuadra. Otro objetivo del guillotinado es obtener varias hojas del mismo pliego o reducir el tamaño para adaptarlos a la guillotina de tres cuchillas.

Las hojas en bruto en lenguaje JDF son un recursos tipo *Media* y los productos finales o intermedios son tipo *Componente*. El proceso de Corte (*Cutting*) genera un recurso de salida del mismo tipo, ya sea la entrada tipo *Media* o *Componente*. El modelo puede verse en la figura 11.2.

Los signos de interrogación en los recursos de entrada indican si son opcionales. En este caso obligan a elegir entre las 2 posibilidades. La salida señalada con el asterisco indica que el número de recursos debe estar entre cero y varios. También debemos advertir que el recurso de salida debe ser idéntico al recurso de entrada y al menos debe ocurrir una vez.

Los datos principales del proceso están definidos en el recurso *Parámetros de Corte* (*CuttingParams*). El corte de los bloques se puede definir en JDF de forma similar a PPF (sección 4.2 figura 4.12). En *CutBlock* solo se definen los cortes, la secuencia se especifica en el recurso *CuttingParams*. La especificación de cada corte es un elemento tipo *Cut*.

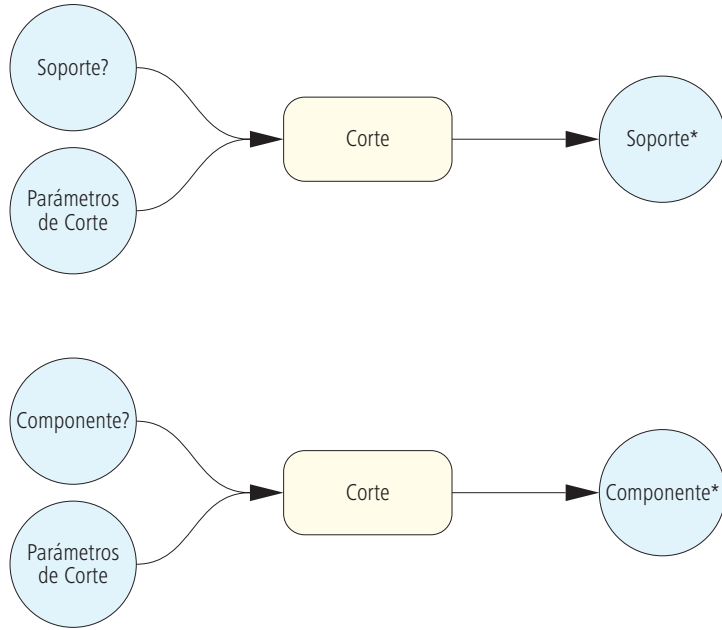


Figura 11.2
Modelo JDF proceso de
corte de las hojas en bruto
(superior)
productos intermedios o
finales (inferior)

En anteriores versiones de JDF las marcas de corte se incluían en recursos *Parámetros de Corte* (*CuttingParams*) pero desde la versión 1.3 todos los elementos de control de impresión se insertan en el recurso *Maqueta* (*Layout*) como describe la figura 10.7.

La figura 11.3 muestra el código del recurso JDF *CuttingParams*; incluye un elemento de corte por bloque (*CutBlock*) y en total son 6. Se ha suprimido algún atributo para hacerlo más legible, los valores en JDF reales tienen más decimales pero aquí los hemos limitado a 2. La posición de cada bloque se define en el atributo *BlockTrf* que se componen de anverso, reverso, lomo y sangrado de 1 cm.

Figura 11.3
Definición de los
bloques de corte

```
<CuttingParams Class="Parameter" ID="_77" SheetName="Cover"
  Status="Available">
  <CutBlock BlockName="Cover_1" BlockSize="768.18 396.84"
    BlockTrf="1.0 0.0 0.0 1.0 96.37 14.17"/>
  <CutBlock BlockName="Cover_2" BlockSize="768.18 396.84"
    BlockTrf="1.0 0.0 0.0 1.0 864.56 14.17" />
  <CutBlock BlockName="Cover_3" BlockSize="768.18 396.84"
    BlockTrf="1.0 0.0 0.0 1.0 96.37 411.02" />
  <CutBlock BlockName="Cover_4" BlockSize="768.18 396.84"
    BlockTrf="1.0 0.0 0.0 1.0 864.56 411.02" />
  <CutBlock BlockName="Cover_5" BlockSize="768.18 396.84"
    BlockTrf="1.0 0.0 0.0 1.0 96.37 807.87" />
  <CutBlock BlockName="Cover_6" BlockSize="768.18 396.84"
    BlockTrf="1.0 0.0 0.0 1.0 864.56 807.87" />
</CuttingParams>
```


El bloque final mide $768.18/396.84 \text{ dtp}=27,1/14,0 \text{ cm}$ y la matriz de valores se define como hicimos en la figura 9.21. La figura 11.4 nos muestra los puntos DTP convertidos en centímetros. El folleto se corta con una guillotina de tres cuchillas mediante un proceso de *Recorte (Trimming)* y se utilizan *Parámetros de Recorte (TrimmingParams)* al formato final de $12,0 \times 12,0 \text{ cm}$. Por lo tanto no se usan el proceso de *Corte (Cutting)*, ni los *Parámetros de Corte (CuttingParams)*.

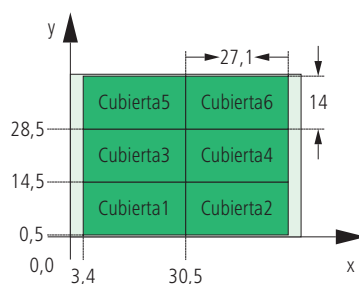


Figura 11.4
Posición del bloque
de corte en la hoja
de impresión

11.2 Plegado

Las máquinas modernas han automatizado la configuración de plegado. Hoy muchos de los ajustes manuales anteriores se insertan en su interfaz y los componentes automotrices adaptan el dispositivo al trabajo, por ejemplo la guía lateral del alimentador, la adaptación de los rodillos al grosor del papel o las paradas programadas para realizar los hendidos. Otros parámetros, no se pueden automatizar como el suministro de aire al separador de hojas, pero podemos crear una "guía" o registro de incidencias provocadas por soportes especiales o ciertos tipos de papel que se puede almacenar en el propio JDF o junto a la información del pedido y recuperar la configuración para una situación similar (quizás exista un software inteligente en el futuro que aprenda, recopile y utilice la configuración adecuada para determinadas condiciones).

Habitualmente la máquina se configura en el menú seleccionando el tipo de plegado e introduciendo el tamaño de la hoja y el sistema electrónico motorizado aplica los ajustes. Tras la primera prueba de plegado se realiza un ajuste fino; este va corrigiendo la desviación de las marcas mediante prueba y repetición. Estas operaciones no suelen provocar distorsiones del papel ni atascos, solo ajustan las pequeñas diferencias relativas a la imposición (pre/post folio) aplicada al plegado, longitud o posición de la pieza.

El ajuste fino no es un valor predeterminado y resulta un poco complicado porque la información procedente del catálogo de plegado no es suficiente para obtener los resultados esperados (figura 11.5) y aunque hayamos incluido la plantilla y el tamaño en el recurso de plegado JDF será necesario reajustar igualmente.

Tenemos conocimiento de algunos sistemas que reducen la tasa de error calculando los pre/postajustes con la secuencia y posición exacta de plegado recibidas directamente en JDF.

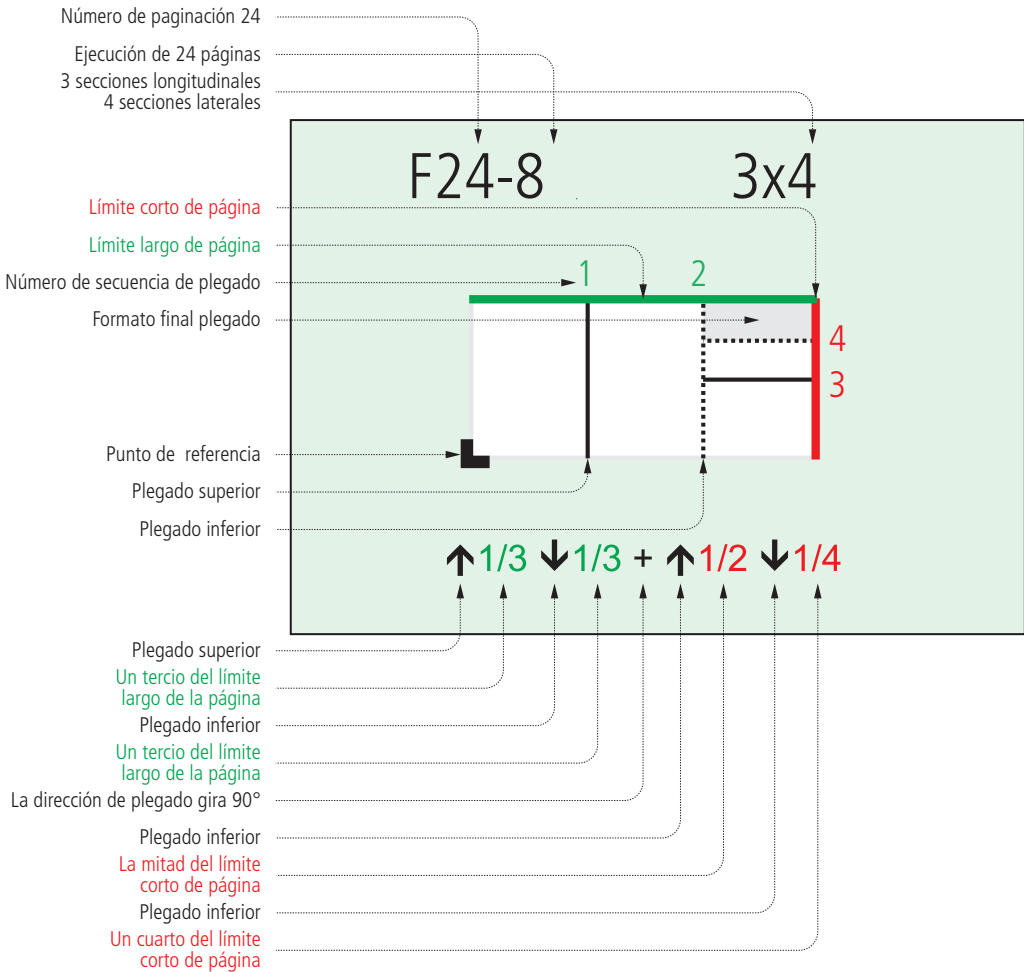


Figura 11.5
Parámetros de Plegado
(FoldingParams) F24-8
de la especificación 1.4

La figura 11.6 muestra el plegado de 24 páginas con el número F-24-8 del recurso *Catálogo de Plegado FoldingParams*. En muchos casos es la única información transmitida.

Figura 11.6
Parámetros de plegado

En el ejemplo se especifica la posición y secuencia del elemento o recurso *Plegado (Fold)*. Si la transferencia del número de catálogo de plegado y las posiciones individuales se realiza desde

```
<FoldingParams ID="FOLD_A-1" Class="Quantity" Status="Available"
  FoldCatalog="F24-8" SheetLay="Left">
  <Fold From="Front" To="Up" Travel="943.9370079811025"/>
  <Fold From="Front" To="Down" Travel="1887.8740161811025"/>
  <Fold From="Left" To="Up" Travel="907.0866141732283"/>
  <Fold From="Left" To="Down" Travel="1360.6299241732283"/>
</FoldingParams/>
```

un Agente o Controlador los elementos se producen directamente en el dispositivo JDF. Este requisito está incluido en la especificación JDF. El atributo *Desde (From)* proporciona la esquina a partir de la cual se produce el doblado y el atributo *A (To)* indica la dirección de plegado.

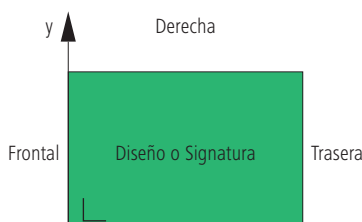


Figura 11.7
Indicación de los límites
de un diseño o signatura

En la figura 11.7 aparecen los límites del diseño o signatura, también podemos ver que el punto de referencia es la izquierda del anverso. Como viene siendo habitual los valores son puntos DTP. La figura 11.5 describe la primera operación de plegado donde la parte delantera coincide con la parte trasera, la hoja se pliega en la dirección y los centímetros indicados en $x=33,3$ cm equivalen a $2.54/943.9370079811025:72$. En la tercera operación de plegado la parte izquierda pliega sobre la derecha $y=32$ cm.

Los objetos que definen el plegado se recopilan desde diferentes módulos según el tipo de trabajo. Si se inserta en el dispositivo JDF un recurso *FoldingParams* como entrada del tipo de plegado, la configuración instalada no lo interpretará y devolverá un mensaje de error.

Los gestores de proyectos o atención al cliente deben conocer la configuración y capacidad del sistema de plegado para determinar la secuencia de entrada de los pedidos. Si se desea implementar una solución de software es muy importante monitorizar el dispositivo JDF porque la combinación de propiedades de un producto condiciona la decisión de plegar en una u otra máquina. Por ejemplo, una máquina puede trabajar con papel de 250 g/m^2 y plegar 6 páginas en Z pero no puede combinar ambas acciones.

La especificación JDF dedica la sección *Capacidad de los Dispositivos (DeviceCapabilities)* a este tema donde se definen valores compatibles, recursos, elementos, nodos, atributos y mensajes JMF.

El MIS o Controlador, por ejemplo, puede transmitir procesos combinados contenidos en *Cajas Grises*, grupos de procesos generales combinados o simples nodos. Las posibilidades pueden ser fundamentales porque se pueden describir detalles JDF más específicos como parámetros de rendimiento, tamaños máximos de hoja permitido e incluso hojas/hora y tiempos estimados de preparación. La figura 7.6 muestra una de estas posibilidades.

11.3 Alzado y Grapado

La línea de alzado y grapado consta habitualmente de 3 módulos:

- La máquina de recopilado.
- La máquina de grapado.
- La guillotina de tres cuchillas.

Existen varios tipos de suministradores para recopilar las diferentes signaturas denominados alzadoras, la mayoría están constituidas por módulos y se alimentan de forma manual o automáticamente mediante varillas o rodillos. El proceso consiste en la inserción de las diferentes signaturas unas dentro de otras hasta formar un bloque, es necesario prestar una atención especial a la secuencia correcta de alimentación (las hojas se insertan desde el interior hacia el exterior). A continuación, los bloques se separan, se abren por la mitad y se colocan sobre una grapadora de lomo. Si el bloque lleva cubierta se inserta en el último módulo, las unidades se apilan una vez terminadas.

La cantidad de signaturas depende del número de alimentadores y del tamaño del producto final. La unidad de grapado se ajusta configurando los alimentadores según el grosor del sustrato y formato de las signaturas.

Una cosedora de alambre o embuchadora es una grapadora que cose los productos por su "espina dorsal", situándolos a horcajadas sobre el lomo y uniendo el bloque mediante grapas de metal. El número, posición, ancho y forma de las grapas puede variar de un pedido a otro. Una bobina suministra el metal al sistema de grapado que consiste en un cabezal que corta el alambre al tamaño adecuado, da forma a la grapa y esta atraviesa la medianil del bloque para doblarse sobre sí misma.

La guillotina de tres lados corta el bloque cosido al formato final y elimina la banda sobrante del plegado para separar las páginas (izquierda, derecha, superior e inferior). Algunas guillotinas disponen de un corte extra central permitiendo la producción de dos unidades simultáneas.

Los 3 procesos de grapado definidos en JDF son: *Recopilación*, *Grapado* y *Recorte*. Generalmente, los módulos de la unidad de grapado están acoplados y se describen como grupo de procesos o procesos combinados.

En la figura 11.8 se representan los recursos de entrada y salida de una *Caja Gris*. A continuación los describimos brevemente:

- El recurso *Dispositivo* (*Device*) identifica la unidad de grapado en línea y su centro de costes.
- En *NodoInfo* se insertan los tiempos de ajuste y producción estimados.
- *Componente* contiene la descripción de la signatura.
- *Ensamblado* es la secuencia de recopilado de signaturas.
- *Parámetros de Grapado* define la forma y número de grapas.
- *Parámetros de Recorte* incluye la información sobre corte y formato del producto final. Sería lógico disponer de un recurso *Parámetros de Recopilación* que describiera la secuencia de selección de las signaturas. De hecho existe uno en la especificación JDF pero no lo explicamos porque es un contenedor de extensiones sin atributos ni subelementos.

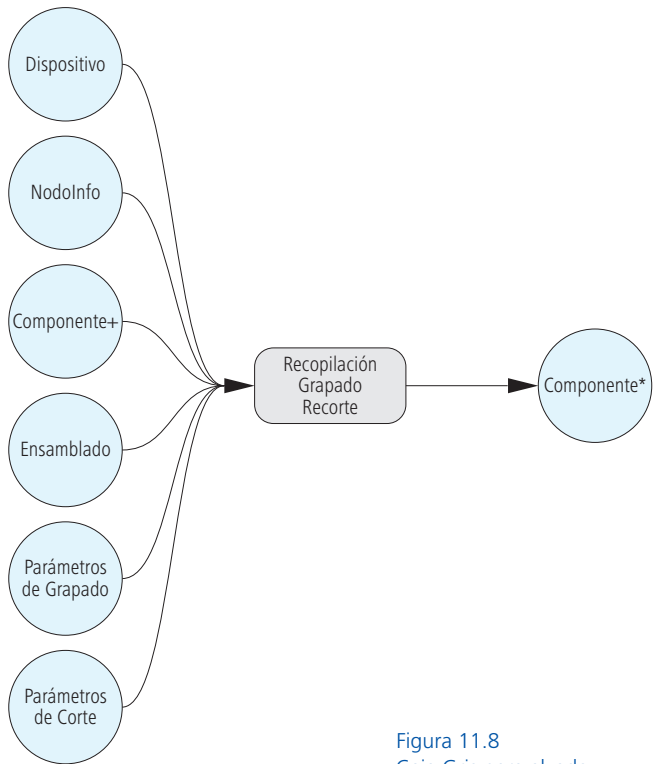


Figura 11.8
Caja Gris para alzado y grapado (recopilado, grapado y recorte de tres lados)

El orden y secuencia de recopilación se especifica concretamente en el elemento *Sección de Ensamblado* (*AssemblySection*) que forma parte de *Ensamblado* (*Assembly*), el primer elemento indica la posición interior y el último la exterior. En la figura 11.9 podemos ver una portada y 3 signaturas.

Cuando la encuadernación es selectiva, el proceso *Recolección* de las signaturas es variable e integra recursos adicionales de entrada en *Normas DB* (*DBRules*) y *Selección DB* (*DBSelection*), permitiendo la adaptación personalizada del producto impreso (diferentes versiones para distintos destinatarios).

```

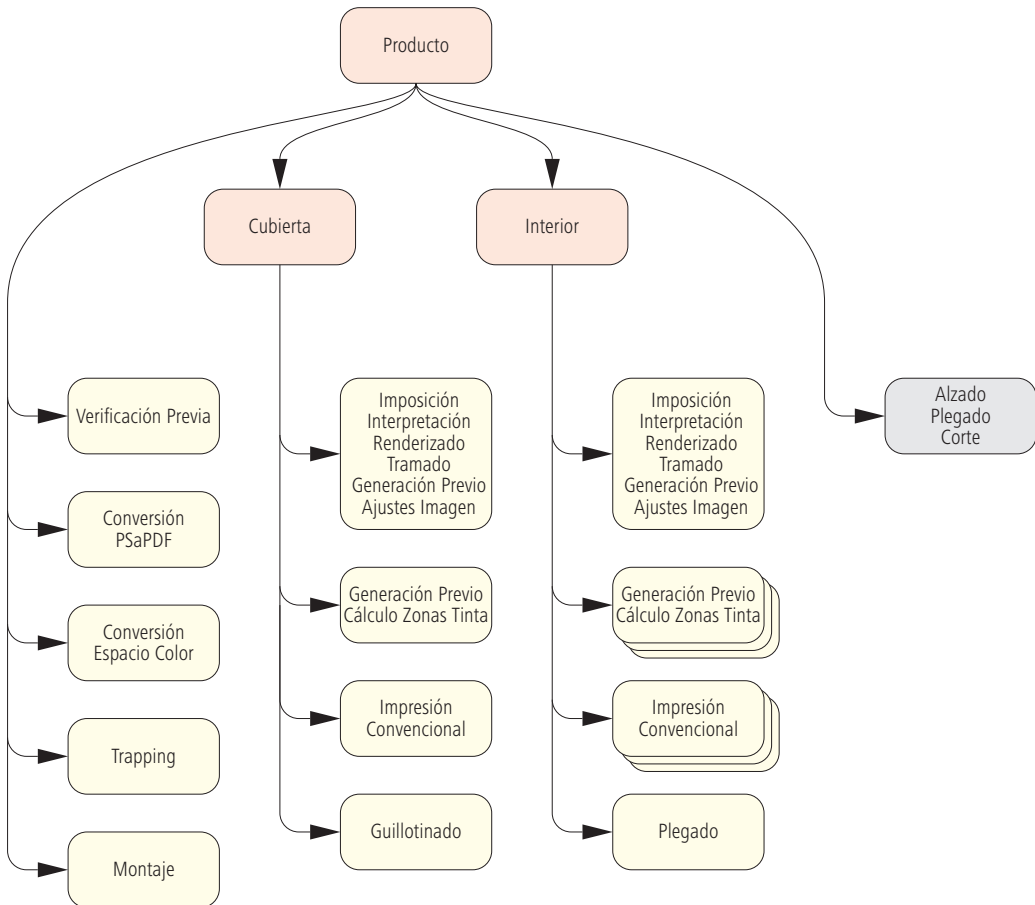
<Assembly Class="Parameter" ID="_019069" Order="Collecting" "
  Status="Available">
  <AssemblySection AssemblyIDs="Cover_B_1"
    DescriptiveName="F04-01_ui_2x1_1" />
  <AssemblySection AssemblyIDs="Text_1_B_2"
    DescriptiveName="F08-07_li_2x2_2" />
  <AssemblySection AssemblyIDs="Text_2_B_3"
    DescriptiveName="F08-07_li_2x2_3" />
  <AssemblySection AssemblyIDs="Text_3_B_4"
    DescriptiveName="F08-07_li_2x2_4" />
</Assembly>

```

Figura 11.9
Secuencia de recopilación
de firmas

Figura 11.10
Árbol del proceso JDF

Para concluir, mostramos el código simplificado del recurso completo *Ensamblado (Assembly)* en la figura 11.9 y en la figura 11.10 representamos el árbol JDF correspondiente. En la columna situada a la izquierda están los procesos básicos de preimpresión: verificación previa de ldatos para producción (*Preflight*), la conversión PostScript a PDF (*PSTToPDFConversion*), conversión entre espacios de



color (*ColorSpaceConversion*), el *Trapping* o *Reventado* y la creación de la maqueta de imposición (*Stripping*).

El procedimiento comienza con la creación de dos *Cajas Grises* para *Preimpresión* e *Imposición*, respectivamente. En este caso no describimos los procesos parciales, las dos columnas centrales muestran los procesos combinados para los subproductos *Cubierta* e *Interior*. Ambos contienen un proceso llamado *Generación de Previo* que cumple diferentes funciones. En el proceso combinado de *Imposición* y *Ajuste de Imagen* su cometido es crear una vista previa para el monitor del panel de la prensa; el compuesto por *Generación de Previo* y *Cálculo de Zonas de Tinta* se utiliza para el preajuste de las zonas de entintado (separaciones a 50,8 ppi). En la parte inferior de las columnas centrales, vemos que tras el proceso de impresión de la *Cubierta* sigue uno de *Corte*, mientras que a *Impresión Convencional* le sigue un proceso de *Plegado* en la columna interior derecha.

La figura 11.11 muestra los recursos de transferencia más importantes del proceso de producción: las páginas para *Imposición* están representadas por recursos *Lista de Ejecución* (*RunList*) y las planchas *Soporte Expuesto* (*ExposedMedia*) son el interfaz entre preimpresión e impresión.

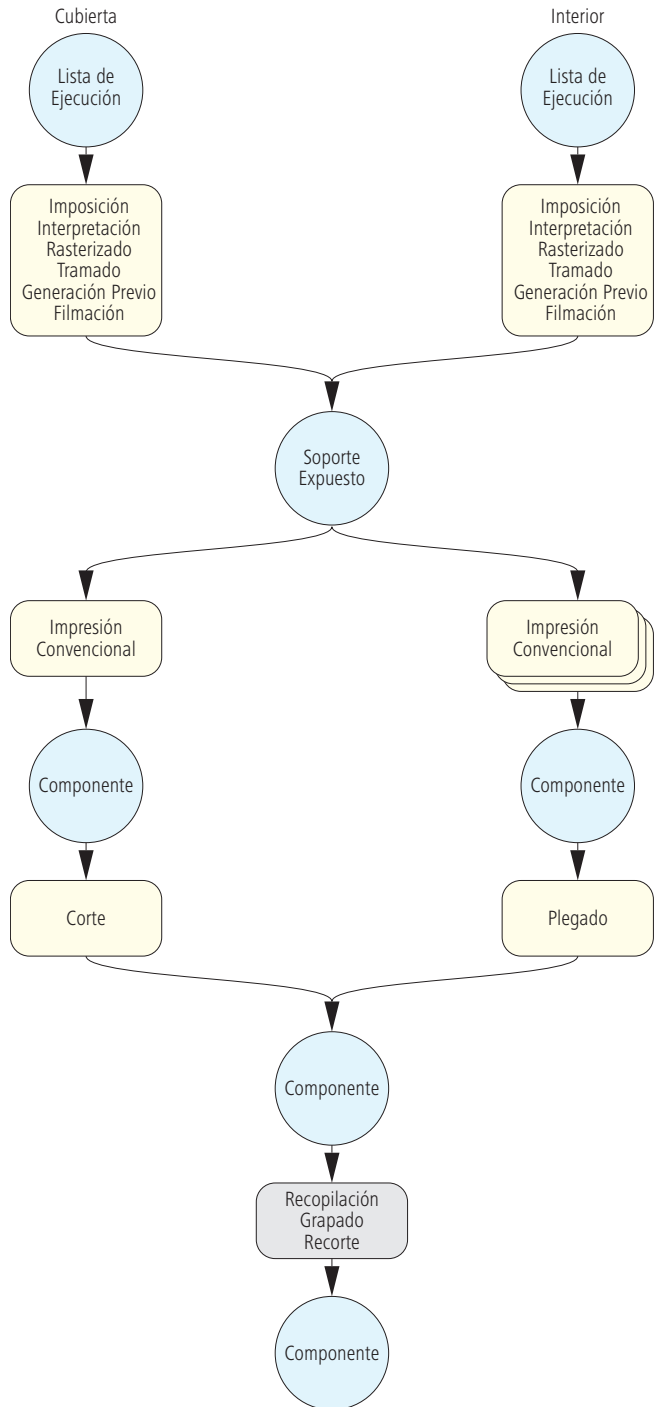


Figura 11.11 Proceso de producción

Por último están los distintos recursos *Componente*: las hojas impresas, las firmas y los productos acabados.

En el capítulo 6 sección 4 de la especificación JDF 1.4, encontramos más de 50 procesos diferentes, pero solo hemos tratado 5 de ellos (Corte, Plegado, Recopilación, Grapado y Recorte). En el capítulo próximo trataremos el proceso *Troquelado (DieMaking)*. Los demás no aportan más claridad al tema y no consideramos oportuno tratarlos.

Ejercicio:

Construir una cadena de procesos similar a la figura 11.11 para un folleto encuadernado con la cubierta realizada por una lámina de oro. Buscar en la especificación JDF la lista de "Recursos de Transferencia" y los procesos necesarios.

Representar un árbol JDF similar al de la figura 11.10 para un folleto encuadernado con los recursos del ejercicio anterior.

12 Impresión de Embalajes

Las técnicas de impresión comercial y packaging no son tan diferentes como cabría esperar, vistas desde la perspectiva del modelo JDF/JMF. De hecho, la impresión de envases utiliza todo tipo de sustratos como los demás sistemas de impresión y acabado (cartoncillo, cartón ondulado, film retráctil o materiales compuestos). El proceso JDF para todos ellos es exactamente el mismo, la única diferencia es la definición y configuración de los recursos, valores y ciertos atributos específicos. Por ejemplo, en el caso del *Tipo de Material (MediaType)* se utiliza considerablemente el cartón ondulado (*CorrugatedBoard*) y el aluminio para envases flexibles metalizados (*Foil*) de uso en alimentación. En recursos tipo *Media* y *MediaIntent* se usan nuevos atributos, un ejemplo es la descripción del canal en cartón ondulado (*Flute*).

Desde JDF 1.4 se ha incrementado el soporte a la impresión flexográfica y se han definido los siguientes valores en el atributo *MediaType*: *Camisa de Flexo (Sleeve)* y *Cinta de Montaje (MountingTape)* y otros alusivos a *Planchas/Clichés (PlateTechnology)* y *Grosor de Relieve (ReliefThickness)*.

En este capítulo presentamos los procesos exclusivos del área de embalaje. Las tres áreas que los rigen son:

- A) Diseño de estructuras para cajas de cartón plegable, optimización del pliegue y troquelado.
- B) Estampado y encolado.
- C) Gestión mediante códigos de barras y compensación del trazo en flexografía.

Figura 12.1
Esquema de tareas para "Cajas de cartón plegables"

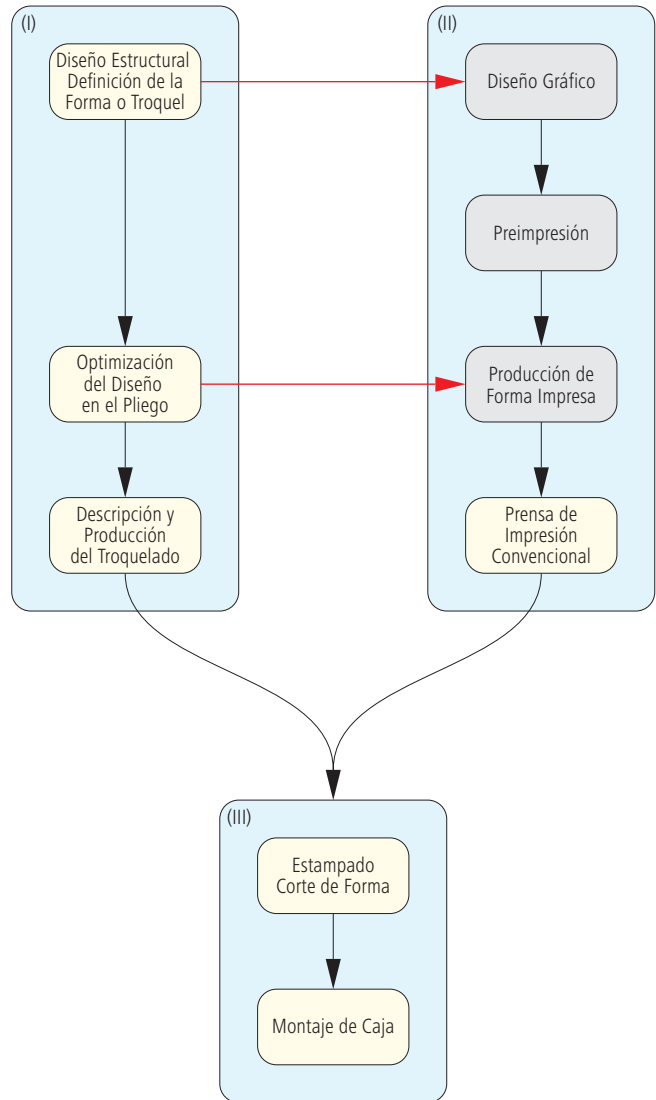
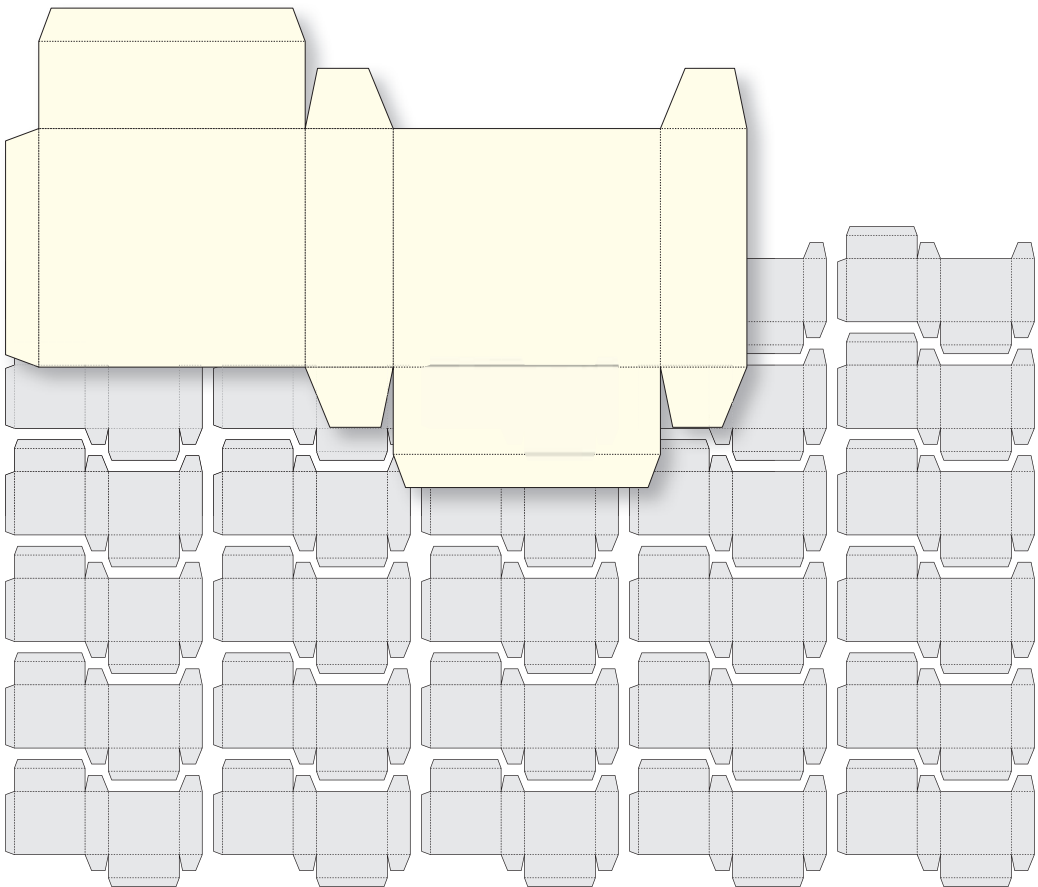


Diagrama de procesos de los ejemplos A y B para producir cajas plegables (figura 12.1).

La compensación del ancho de línea es un proceso de Preimpresión parcial. En el gráfico vemos las actividades correspondientes a procesos JDF destacadas con rectángulos amarillos. Normalmente, la forma se diseña en un sistema CAD donde se define el contorno, marcas y líneas de perforado. La optimización del diseño en pliego (*DieLayoutProduction*) intenta reducir el desperdicio de material situando el máximo número de unidades en la superficie disponible y al mismo tiempo facilita la ejecución acoplando unos diseños en otros como muestra en la figura 12.2. Este proceso se suele realizar en el sistema CAD junto a la colocación de los agujeros de registro, imprescindibles para la máquina de troquelar. Los datos CAD exportados se pueden importar de nuevo al sistema correspondiente para producir los troqueles. La figura 12.3 muestra distintos componentes de un troquel:

Figura 12.2
Unidad y desarrollo
optimizado de un pliego

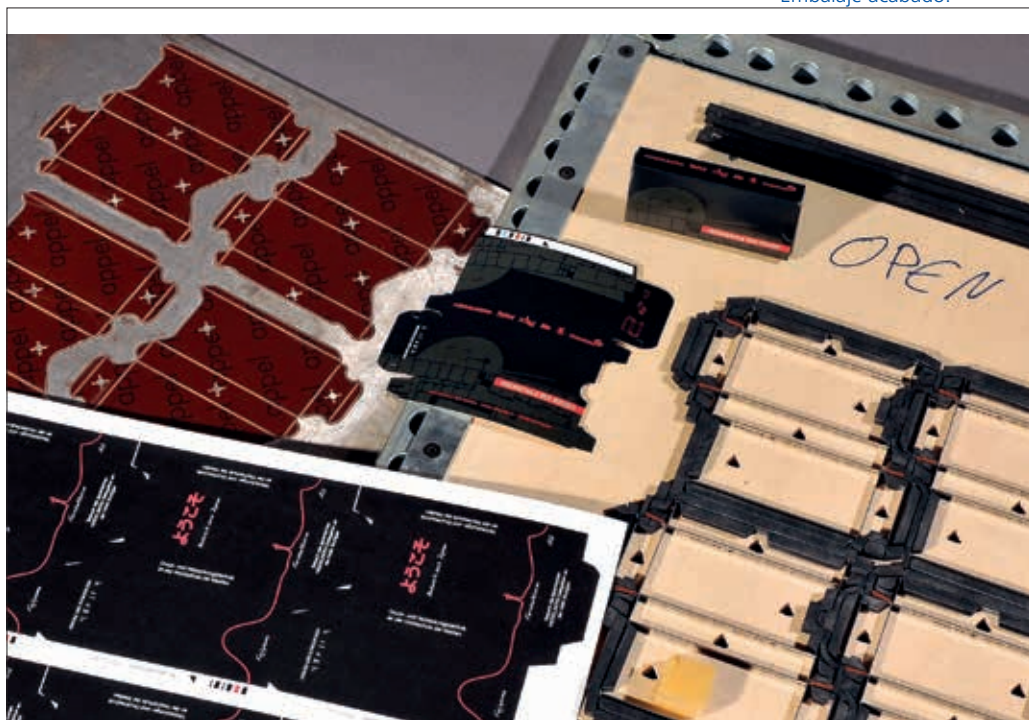


- Troquel de corte (*CutDie*).
- Contra matriz (*CounterDie*).
- Molde y Matriz de Separación (*Stripping Tool, Lower&Upper*).
- Cauchos (*Blanking Tool, Blanker*).

La contra matriz crea los hendidos y los cauchos separan el residuo de la perforación. Las *Formas (Tools)* para *Troquelar (ShappeCutting)* se realizan en *Producción de Troqueles (DieMaking)*. Se montan en el bastidor y este se coloca en la máquina de estampación para cortar y separar las piezas del pliego. El troquelado puede ser independiente de la separación, una máquina troquela las piezas y otra las separa. Por último, las piezas pasan a un sistema en línea denominado encoladora que pliega, encola y cierra las cajas de cartón. El proceso combinado se denomina *Montaje de Caja (BoxFolding)*.

La figura 12.1 muestra el flujo de datos específico mediante las líneas rojas. La información del troquel y los hendidos ajustan el diseño gráfico al diseño estructural de la caja de cartón plegable. Habitualmente se exporta un archivo EPS desde el sistema CAD para guiar al diseñador o se inserta la referencia al archivo PDF.

Figura 12.3
Elementos de troquelado:
troquel en bastidor (dcha.),
contra matriz (superior izq.),
pliego impreso (inferior izq.),
unidad troquelada (centro),
Embalaje acabado.



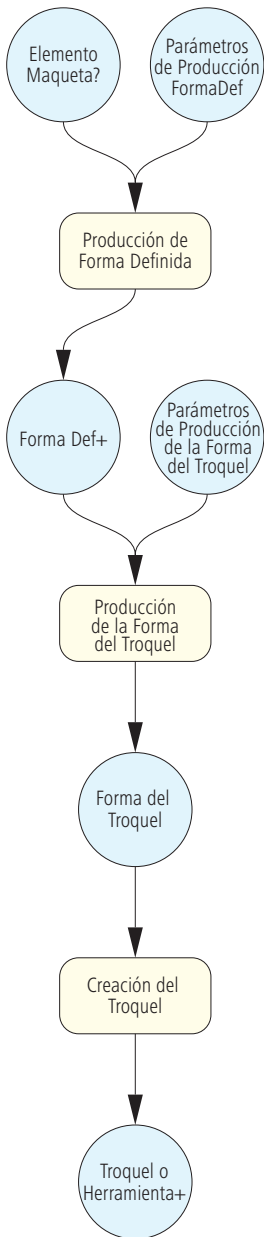


Figura 12.4
Diagrama para la
creación de un troquel

Habitualmente, el sistema CAD diseña la forma y posiciona las unidades hasta cubrir toda la superficie del pliego, mediante la función de imposición *Paso y Repetición (Step&Repeat)*, coincidiendo perfectamente con las formas de troquelado y hendido. Los datos se transmiten al fabricante del troquel y al impresor en formato DXF, CFF2 o EDDA. La producción del troquel es muy laboriosa y la suele realizar una empresa externa. Esta recibe los dibujos CAD al inicio para que las planchas estén listas en la fecha prevista. Por esta razón, el impresor no determina la posición de las unidades en el pliego. El MIS proporciona la información de repetición al troquelador y al preparador de la forma de impresión mediante *Etiquetas (Labels)* delimitadas por rectángulos (*BoundingBox*). El software de producción o el sistema CAD organiza el plegado y montaje de las cajas de cartón más complejas optimizando las unidades sobre el pliego.

12.1 Diseño y fabricación del Troquel y Optimización del Pliego

La figura 12.1 cuadro (I), muestra el diseño estructural y los datos CAD para fabricar el troquel, el recuadro (II) muestra diseño, preimpresión e impresión. El recuadro (III) representa las tareas de acabado. Los procesos del cuadro (I) son recientes, se incluyeron por primera vez en la especificación JDF 1.4, por lo tanto, los ejemplos expuestos no están basados en datos de producción. Hace años que se realizan instalaciones con procesos JDF como los de (II) y (III) aunque son escasas. En la figura 12.4 representamos el cuadro correspondiente al (I) de la figura 12.1, pero no se muestra como modelo de proceso/recurso JDF. Para la *Producción de la Forma (ShapeDefProduction)*, por ejemplo la creación CAD de una caja de cartón, etiqueta, expositor, bolsa, etc, disponemos de 2 recursos de entrada, en uno se introduce un elemento *Maqueta (LayoutElement)* que este puede ser la dirección URL del boceto EPS realizado en CAD, procedente de la agencia de publicidad o departamento de diseño del cliente. En el otro recurso, *Parámetros de Producción de la Forma (ShapeDefProductionParams)* se puede incorporar la dirección URL del modelo en 3D del embalaje. También se pueden añadir las normas estandarizadas de producción de los catálogos de ECMA, AFCO o FEFCO.

La salida *Producción de Forma Definida Forma Def+*, describe la construcción de la caja con una referencia al archivo que contiene los contornos o introduciendo estos en el propio recurso, en este caso las rectas y curvas Bezier se incluyen en el atributo *CutPath* como valores codificados igual que en PDF (sección 4.4.1 [5]).

La figura 12.5 describe ambas posibilidades, la referencia al archivo CFF2 y la descripción del recurso *Forma* (*Shape*), es decir, el trazado se define en el recurso *Trazado de Recorte* (*CutPath*) como en PDF: El operador *m* especifica un punto, el operador *l* dibuja una línea desde ese punto, finalmente, el operador *c* dibuja una curva a continuación de la línea (*m = MoveTo*, *l = LineTo*, *c = CurveTo*). Los operandos siempre se sitúan delante del operador. La norma ANSI DDES3 [7] que lo fundamenta se define en el atributo *Tipo de Corte o Perforación* (*DDESCutType*).

Figura 12.5
Dos formas posibles para describir los contornos de las cajas de cartón plegable

```
<ShapeDef Class="Parameter" ID="_4711" Status="Available">
  <FileSpec URL="file://Fileserver1/CFF2/Folding.cf2"/>
</ShapeDef>
```

```
<ShapeDef>
  <Shape ShapeType="Path" DDESCutType="101"
    CutPath=" 28 28 m 10 72 1 20 140 144 150 144..." />
</ShapeDef>
```

Si el envase está compuesto por varias partes (una caja con tapa) se añade al proceso *Producción de Maqueta* (*LayoutProduction*) un recurso *Definición de Forma* (*ShapeDef*). Los envases adicionales se ordenan y distribuyen sobre el pliego mediante un recurso *Producción Forma del Troquel* (*DieLayoutProductionParams*) y parámetros de repetición (*Step&Repeat*). A continuación, el proceso *Creación del Troquel* (*DieLayoutProduction*) dispone los contornos para optimizar el pliego con los datos CAD de la forma y la matriz de imposición en el recurso *Forma/Desarrollo del Troquel* (*DieLayout*), básicamente suele ser la referencia al archivo externo. Luego se fabrica la herramienta física (*Tool*) descrita en el proceso *Creación del Troquel* (*DieMaking*).

Aparte de su uso para realizar troqueles este método también se utiliza para fabricar herramientas ciegas de separación e instrumentos de corte, estampación, relieve, golpe en caliente y contra troqueles, pero no los trataremos en el texto.

El flujo de producción gráfica se completa con la imposición y filmación de las planchas para producir las formas impresas como contrapartida a la creación de los troqueles (*DieLayout*).

El proceso *Parámetros de Montaje* (*StrippingParams*) ya ha sido tratado en los capítulos 8-9 y las figuras 8.7, 8.9 y 9.8. La posición de plegado de una hoja respecto a las demás en impresión comercial, es fija y se registra en sus subelementos (figura 8.9).

Por el contrario, los datos de contenido en la producción de envases se organizan con una matriz que repite los contornos optimizados (*Step&Repeat*) en lugar del esquema de imposición.

Algunos sistemas MIS son capaces de transmitir la información procedente del departamento de CAD, aunque todavía no utilizan JDF para este propósito.

El recurso *Signatura de Encuadernación* (*BinderySignature*) de la figura 12.6 muestra el código JDF. Los datos CAD se conectan con una referencia URL. Primero se sitúan las líneas de troquel sobre el pliego con *StrippingParams*, la única diferencia con el ejemplo de la figura 8.8 es que allí habían más firmas. También podemos ver que contiene el subelemento *StripCellParams* como ocurría en el ejemplo anterior. En él se define el formato final de la caja de cartón y se adjunta la referencia al trazado (*Mask*) que es idéntica al contorno de corte del troquel.

El recurso *BinderySignature* puede definirse en el recurso hijo o de forma independiente en *StrippingParams* (figura 8.9). En este último caso el recurso *StrippingParams* debe contener una referencia a *Parámetros de Encuadernación* (*BinderyParams*) como vemos en la figura 12.6.

Las *Signaturas de Encuadernación* (*BinderySignature*) se dividen en tres tipos (*BinderySignatureType*):

- *Patrón de Plegado* o *Folding Pattern* (*Plegado-Fold*).
- *Paso y Repetición* o *Step&Repeat* (*Matriz-Grid*).
- *Trazados de Troquel* o *DieLine* (*Troquel-Die*).

Figura 12.6
Optimización de la
distribución mediante
datos CAD

```

                <StrippingParams Class="Parameter" ID="_001"
Status="Available" WorkStyle="Simplex">
  <Position MarginBottom="36.0" MarginLeft="36.0"
  RelativeBox="0.00000 0.00000 1.00000 1.00000" />
  <BinderySignatureRef rRef="_4711" />
  <StripCellParams Mask="DieCut" TrimSize="744.12 752.04" />
  ...
</StrippingParams>

<BinderySignature BinderySignatureType="Die" Class="Parameter" ID="_4711"
Status="Available">
  <DieLayout>
    <FileSpec URL="file://Fileserver1/DDES3/6-Folding.dd3" />
    <Station StationAmount="6" StationName="DES1" />
  </DieLayout>
</BinderySignature>

```

Hasta ahora no ha sido necesario mencionar el atributo *Plegado* (*Fold*) pero aquí es el valor predeterminado. Si el recurso es de tipo *Troquel* (*Die*), debe incluir una referencia/dirección URL al elemento ("6-foldbox.dd3") que contiene la *Línea de Troquel* (*DieLayout*). En el ejemplo también se define un subelemento *Station* que indica el número de unidades desplegadas sobre el pliego de impresión.

La figura 12.7 muestra un ejemplo del recurso *BindarySignature* en el cual se define la disposición de la unidad mediante una matriz *Paso y Repetición* (*Step&Repeat*). El resultado es una nueva matriz tipo *BindarySignature* donde el atributo *NumberUp* limita el número de columnas y filas, de modo que cada entrada de la matriz corresponde a un corte de la caja de cartón plegable. En impresión comercial se utiliza el mismo atributo *NumberUp* para definir el patrón de imposición, las entradas de la matriz corresponden al patrón de cada página (ver figura 8.11 sección 8.2). La orientación de cada hoja/unidad se indica en el atributo *FrontPages*, el valor 0 representa la misma orientación para todas las etiquetas del pliego y además sabemos que son iguales porque de no ser así se habría definido más *Signaturas de Encuadernación* (*BinderySignatures*).

Figura 12.7
Step&Repeat para etiquetas

12.2 Perforado y Encolado de Cajas de Cartón

```
<BinderySignature ID="_4712" Status="Available" BinderySignatureType="Grid"
  NumberUp="2 3"...>
  <SignatureCell FrontPages="0 0 0 0 0 0 " Orientation="Left" />
</BinderySignature>
```

Habitualmente los pliegos de impresión se troquelean en máquinas de perforación automáticas con piezas realizadas a medida, denominadas matrices de corte o troqueles. Tras la perforación y según sea la configuración, la tarea de separar los residuos puede ser en línea o realizarse en una estación independiente. Al finalizar las unidades quedan cortadas y separadas.

El recurso JDF, *Corte de Forma* (*ShapeCutting*) describe estas operaciones. Los elementos más importantes son la *Herramienta* (*Tool*), tratada en el capítulo anterior y el *Componente* hoja de impresión.

La figura 12.8 representa este modelo. Podríamos integrar otros recursos de entrada tipo *NodeInfo* o *Device*. Además si se pueden hacer relieves o estampados en caliente deberíamos indicarlo integrando otras máquinas o automatizando la máquina de perforar. Serían procesos combinados que incluirían *Relieve* (*Embossing*) y *Corte de Forma* (*ShapeCutting*).

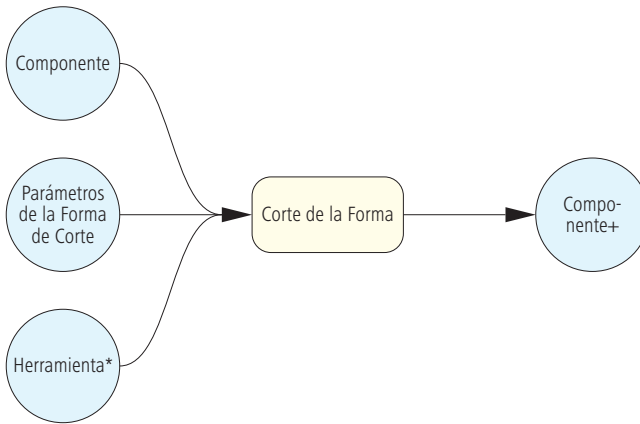


Figura 12.8
Modelo JDF de troquelado/
perforado/estampado

Ya vimos el proceso *Corte de Forma (ShapeCutting)* en el modelo de la figura 12.8, la figura 12.9 muestra un fragmento del mismo código simplificado, es decir, no hay recurso de entrada *Herramienta (Tool)*. Los sistemas de producción actuales no facilitan esta información como explicamos al inicio de la sección 12.1. En impresión comercial, un *Nombre de Bloque (BlockName)* se identifica con el elemento *Corte de Bloque*

(*CutBlock*) para un proceso de *Corte (Cutting)* (figura 11.3), pero en impresión de embalajes se identifica como unidad troquelada. El proceso *ConventionalPrinting* produce un *Componente* que representa una pila de *Cartón Plegable/Pliegos Impresos* con el *Nombre (SheetName)* del cual se troquelarán las 6 cajas de cartón plegables (*Blocks*).

La producción de las cajas se inicia introduciendo el material cortado en el alimentador de la máquina de encolar, se realizan los hendidos por las líneas de plegado, se doblan y se vuelven a extender en plano para facilitar la apertura de las cajas en la máquina de embalaje, finalmente se encolan las pestañas, se pegan y se cierran las cajas.

Ciertas cajas de cartón encolado se realizan a la medida solicitada por el cliente y también hay modelos especiales, por ejemplo, las cajas de chocolate o moldes que producen formas cónicas. Estos casos se realizan a partir de componentes fabricados por distintos proveedores incluso se construyen máquinas a medida para el propósito previsto.

Por regla general, para adaptar la encoladora al proceso, el usuario debe realizar muchos ajustes mecánicos y debe anotar los valores para casos de repetición. Sin embargo, algunas máquinas almacenan y aplican automáticamente estos ajustes mediante software y servomotores sin necesidad de utilizar llaves, ruedas o palancas, como ocurre con las máquinas tradicionales.

A continuación, describimos como ajustar una máquina de encolado. Primero se adapta el alimentador según el formato del trabajo y tipo de corte. Después se configuran las unidades de extrusión adaptándolas a la forma del producto final y el grosor del material.


```

<JDF ID="_343" JobPartID="_1002.0" MaxVersion="1.3" Status="Part" Type="
ShapeCutting " Version="1.3" >
  <ResourceLinkPool>
    <NodeInfoLink Usage="Input" rRef="_344" />
    <ComponentLink Usage="Input" rRef="172"...>
    <DeviceLink Usage="Input" rRef="_339" />
    <ComponentLink Usage="Output" rRef="_152"...>
    <ShapeCuttingParamsLink Usage="Input" rRef="_772">
      <Part SheetName="Folding Sheet" SignatureName="SIG1" />
    </ShapeCuttingParamsLink>
  </ResourceLinkPool>
  <ResourcePool>
    <ShapeCuttingParams Class="Parameter" ID="_772"
      PartIDKeys="SignatureName SheetName BlockName" SheetLay="Left"
      Status="Available"...>
      <ShapeCuttingParams SignatureName="SIG1">
        <ShapeCuttingParams SheetName="Folding Sheet">
          <ShapeCuttingParams BlockName="Block1" />
          <ShapeCuttingParams BlockName="Block2" />
          <ShapeCuttingParams BlockName="Block3" />
          <ShapeCuttingParams BlockName="Block4" />
          <ShapeCuttingParams BlockName="Block5" />
          <ShapeCuttingParams BlockName="Block6" />
        </ShapeCuttingParams>
      </ShapeCuttingParams>
    </ShapeCuttingParams>
  </ResourcePool>
  ...
</JDF>

```

El paso siguiente es adaptar la unidad de encolado. Aquí se define la longitud, anchura y posición de las unidades a encolar y la tasa de aplicación de la tira de adhesivo.

Figura 12.9
Proceso de troquelado/
perforado/estampado

Las unidades de encolado requieren un tratamiento específico de la presión, el plegado posterior y la recolección de las cajas terminadas.

La integración de una máquina encoladora de cartón en una red JDF/JMF aporta beneficios similares a las demás máquinas:

- Los datos de la orden llegan directamente a la máquina.
- Los datos de manufactura se transmiten directamente al sistema de producción.
- Se almacenan los datos de ajuste por defecto reduciendo los tiempos de preparación.

El tercer punto tiene una incidencia menor porque el preajuste automático no es habitual en las máquinas existentes en el mercado pero cabe destacar las posibilidades de JDF de cara al futuro.

El trabajo de la máquina encoladora se representa en *Plegado de Caja (BoxFolding)*. Los valores de recorte son recursos de entrada tipo *Components* procedentes del proceso de estampación, los datos contenidos en *BoxFoldingParams* pueden ser interpretados por la máquina (ver ejemplo figura 12.10). En el código reconocemos las *Acciones de Plegado de Cajas (BoxFoldActions)*. En la especificación JDF existen 10 normas predefinidas pero se pueden definir cajas de cartón personalizadas.

En la figura 12.11 vemos el *Tipo de Plegado de Caja (BoxFoldingType)*, correspondiente a la construcción de un modelo *Type01* y los valores en puntos DTP del atributo *Dimensión del Soporte X/Y (BlankDimensionsX/BlankDimensionsY)*. El origen o punto cero está situado en la parte inferior izquierda.

Este caso describe una caja del tipo indicado, obviamente, los valores de otros modelos serán distintos. Al incorporar el tipo de caja se informa simultáneamente de las operaciones de plegado asociadas.

Es importante indicar que los recortes se establecen en el alimentador con la cara impresa hacia abajo:

Figura 12.10
Datos predefinidos para
una máquina de encolar
cajas de cartón

- *Doble* en línea 0 de izquierda a derecha (*Creased*).
- *Doble* en línea 2 de derecha a izquierda (*Creased*).

```
<BoxFoldingParams BlankDimensionsX="43 388 502 848 960"
BlankDimensionsY="43 78 156 465 543 578 621" BoxFoldingType="Type01"
Class="Parameter" ID="_771" PartIDKeys="SignatureName SheetName BlockName"
Status="Available">
  <BoxFoldingParams SignatureName="SIG1">
    <BoxFoldingParams SheetName="Folding Sheet">
      <BoxFoldingParams BlockName="Block1" />
      <BoxFoldAction Action="LongPreFoldLeftToRight"
FoldIndex="0 -1" />
      <BoxFoldAction Action="LongPreFoldRightToLeft"
FoldIndex="2 -1" />
      <BoxFoldAction Action="LongFoldLeftToRight" FoldIndex="1 -1" />
      <BoxFoldAction Action="LongFoldRightToLeft" FoldIndex="3 -1" />
    </BoxFoldingParams>
  </BoxFoldingParams>
</BoxFoldingParams>
```

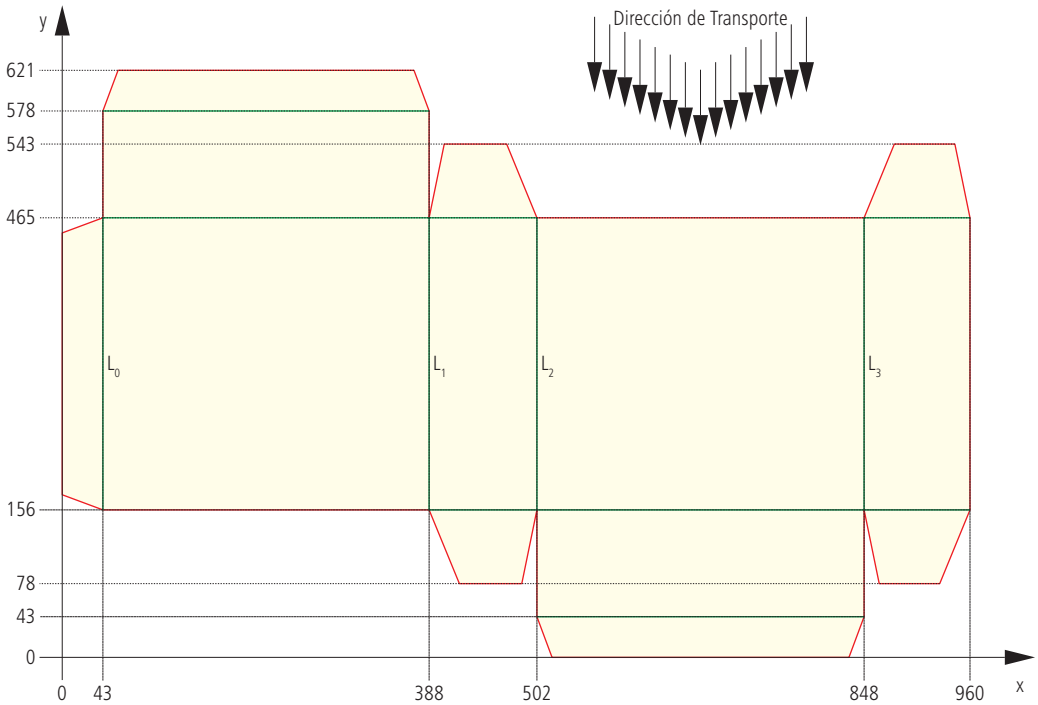


Figura 12.11
Forma estándar Tipo 01
de caja de cartón plegable

Una vez aplicada la cola desde abajo sobre la pestaña izquierda:

- Plegar sobre la línea 1 de izquierda a derecha (*Folded*).
- Plegar sobre la línea 3 de derecha a izquierda (*Folded*).

La franja de encolado (*GlueLine*) no está definida en el ejemplo pero el elemento *BoxFoldAction* corresponde exactamente al orden de las operaciones de plegado. El único valor crítico del indicador de plegado (*FoldIndex*) es el primero "-1" y nos comunica que el plegado es longitudinal en la dirección de desplazamiento.

12.3 Códigos de Barras

A primera vista, el código de barras es un gráfico más del envase y lo natural es incluirlo durante el proceso de diseño. De hecho los códigos de barras se crean con los programas de diseño habituales; no queremos decir que la generación de códigos de barras forme parte de los programas. Son añadidos mediante la instalación de programas de tercera partes al efecto (*pluggings*). Existen aplicaciones especializadas independientes que instalan una tipografía en el sistema operativo e insertan el código de barras correspondiente mediante un carácter individual predefinido.

```

<LayoutElementProductionParams Class="Parameter" ID="_300" Status="Available">
  <LayoutElementPart>
    <BarcodeProductionParams>
      <IdentificationField Encoding="BarCode" EncodingDetails="EAN_13"
        Value="0123456789128" />
    </BarcodeProductionParams>
  </LayoutElementPart>
</LayoutElementProductionParams>

```

Figura 12.12
Definición de
código de barras

Una vez realizado el código se coloca en el diseño del programa de maquetación. En el sector del embalaje es normal trazar todas las fuentes, por lo tanto, el código de barras se incorpora al diseño como un gráfico más, sin necesidad de facilitar la tipografía específica al proveedor de impresión. Esta práctica tiene un gran inconveniente que inhabilita el código, la impresión en flexografía produce un reventado en las líneas que impide la lectura con dispositivos electrónicos. Este tipo de error tiene unas consecuencias enormes, deja el envase inservible e inutiliza el producto si el error se advierte tras el envasado.

El cálculo de compensación del ancho de línea del código de barras se debe realizar antes de incluirlo en el diseño final. Este valor depende de diferentes factores: tecnología de impresión, sustrato, tinta, residuo, etc y el cumplimiento de estos requisitos debe ser supervisado por técnicos especialistas en preimpresión de envases, por lo tanto no se pueden incorporar en el diseño gráfico inicial. Por norma los códigos de barras se deben añadir a las planchas por los servicios de impresión mediante programas especiales para preimpresión de embalajes.

La información facilitada por el cliente referente a los códigos de barras debe transferirse a producción y JDF facilita esta comunicación. El impresor recibe el código EAN* como una secuencia de números que introduce en el MIS. Una vez allí el empleado no necesita volver a insertarla, se transmite al software de preimpresión de embalajes mediante JDF y se evita el riesgo de cometer errores de transcripción.

La entrada correspondiente se encuentra en la figura 12.12. El elemento *Parámetros de Producción de Elementos de Maqueta* (*LayoutElementProductionParams*) es el recurso de entrada del proceso *LayoutElementProduction*.

* EAN es un sistema internacional de código de barras utilizado en la venta de productos al por menor. En U.S.A. y Canadá se utiliza el sistema de códigos de barras UPC. El código EAN es similar a UPC excepto en que utiliza 13 dígitos.

Las unidades o páginas completas del programa de diseño, ya sean imágenes, gráficos, textos o códigos de barras, se crean y describen en el proceso *LayoutElementProduction*. El MIS no detalla el proceso, produce una *Caja Gris* o un proceso combinado [descritos anteriormente en *Preparación de Preimpresión (PrePressPreparation)* y *Creación de Contenido (ContentCreation)*]. El parámetro para generar los códigos de barras se incluye en el subelemento *Parámetros de Producción de Códigos de Barras (BarcodeProductionParams)*. Cada código figura en su *Campo Identificativo (IdentificationField)*. En el presente ejemplo aparecen el *Valor (Value)* y el *Tipo de Código de Barras (EncodingDetails)*. Este recurso puede contener subelementos en forma de recursos adicionales y referencias a recursos, en la figura 12.13 por ejemplo, vemos la altura y anchura del código de barras (*BarcodeReproParam*) o la compensación del ancho de línea (*BarcodeCompParams*).

Una ventaja añadida de la inserción de códigos de barras mediante metadatos es la garantía de calidad, se puede corroborar su coincidencia comparando el valor con una base de datos de producto. Si se hubiera incluido un objeto gráfico no sería posible este tipo de control de calidad.



Figura 12.13
 “Árbol de recursos” para el proceso Producción de Elemento de Maqueta (*LayoutElementProduction*)

13 Proyectos JDF/JMF

Sería razonable pensar que los hechos respaldan los últimos doce capítulos pero nos movemos sobre una delgada capa de hielo. Cada caso es único y los métodos para gestionar proyectos JDF siempre son distintos. Por esta razón resulta tan complicado sugerir sobre pautas generales para implementar un flujo de trabajo JDF. Además, en cierta medida las observaciones expresadas en este escrito se basan en nuestra propias experiencias. No queremos dar una imagen extremadamente pesimista y tampoco deseamos expresar un optimismo excesivo. Para concluir y ser prudentes será mejor que simplifiquemos la integración de proyectos JDF en:

- Flujo de trabajo JDF con módulos de un proveedor.
- Flujo de trabajo JDF con módulos de múltiples proveedores.
- Flujo de trabajo con módulos de varios proveedores y adaptación de un programa patentado.

Antes de profundizar en los casos debemos realizar ciertas observaciones pero no confundir con el sermón del domingo:

“Ante todo y especialmente al iniciar la implementación del flujo de trabajo JDF debemos reducir nuestras expectativas al mínimo”.

Es decir, lo que podemos esperar de un flujo de trabajo JDF, en gran medida, depende de las exigencias de cada operación. Cuanto más desconocemos el tema más difícil nos resultará evaluar la situación porque la complejidad está en los pequeños detalles. En ocasiones se espera algún tipo de milagro, por ejemplo, si la dirección de la empresa trata de controlar la aplicación del proyecto o los empleados de impresión no son capaces de advertir lo que está sucediendo realmente y se muestran decepcionados por el control. “Flujo de Trabajo JDF” pueden surgir comentarios del tipo: ¿con recibir un par de valores por defecto ya está? o ¿eso es todo?

Ciertamente, algunos proveedores ya ofrecían anteriormente algunas de las soluciones que se plantean con JDF/JMF, como la captura de datos en planta o el intercambio de órdenes y ajustes preestablecidos entre departamentos, sin embargo, consideramos que sus capacidades no están agotadas.

El concepto JDF/JMF supera con creces las soluciones facilitadas por los proveedores y es una tecnología perfectamente factible en la actualidad y con un gran potencial orientado al futuro.

Para aclarar las expectativas podemos expresarlo con otras palabras, debemos fomentar los argumentos que inciten a la participación en el proyecto JDF/JMF. Para ello debemos profundizar en temas como la transparencia y simplificación de la comunicación global (interna/externa), la reducción de errores mediante automatización o mejora de la producción, la planificación sistematizada y la reducción de costes y residuos. Quizás el mejor argumento sería hablar de todo ello en conjunto. Pero siempre surge el mismo problema, las razones resultan abstractas, las esperanzas demasiado vagas y los objetivos difíciles de cumplir, son palabras que ya hemos oído antes.

La base para evitar esta situación es realizar un profundo análisis de la situación real y definir los objetivos técnicos y empresariales con claridad, examinar el flujo de trabajo actual es valioso incluso para desechar el uso de JDF. Descubrirá nuestras deficiencias desde un punto de vista totalmente diferente y nos puede guiar hacia las soluciones correspondientes. Pero, si nos centramos en el compromiso JDF, el análisis revelará las metas reales de nuestro proyecto. Para sintetizar, la imprenta debe redactar una memoria de sus necesidades funcionales y confirmar que los posibles proveedores de soluciones JDF pueden cumplir sus expectativas y compromisos.

La implementación JDF/JMF es un proyecto “por etapas” a largo plazo. En principio se trata de trabajar con eficacia y los beneficios llegarán posteriormente. Por lo tanto, el proyecto se debe abordar en conjunto y de común acuerdo definiendo fases y objetivos intermedios. Los pasos, que concretaremos posteriormente, se puede dividir en los siguientes puntos generales:

- Creación de la red de MIS a preimpresión.
- Red de MIS/preimpresión a impresión.
- Captación de datos en producción y transferencia al MIS.
- Integración de las máquinas de acabado en la red.
- Conexión con el cliente.

Por norma el MIS es la base de una red JDF. Los pasos siguientes dependen en gran medida del equipamiento disponible. Desde el punto de vista económico por ejemplo, no parece muy lógico adquirir una nueva plegadora con motor servo porque encaja en el entorno JDF. Resulta más razonable actualizar las máquinas existentes para que dispongan de un interfaz JDF compatible o valorar si la compatibilidad JDF justifica la sustitución.

Los 3 casos presentados necesitan la admisión del nuevo sistema de trabajo mediante la participación activa de los usuarios. Nos gustaría citar las palabras de James Harvey, director ejecutivo de la Organización CIP4 [21]:

“Implementar un proceso de automatización, incluso con JDF, necesita la participación activa del personal de la imprenta. Los proveedores basan la automatización de la imprenta y sus procesos en diferentes filosofías y el impresor, en calidad de administrador del proyecto debe encontrar la forma de resolver los problemas que puedan surgir cuando se integran múltiples sistemas”

Por lo tanto, la participación no debe ser a una orden de “arriba”. El ambiente de trabajo debe caracterizarse por la ilusión que provoca la innovación y el espíritu de equipo. Los sistemas de flujo de trabajo (con o sin JDF) deben transformar los límites de responsabilidad interdepartamental. La creación de un entorno JDF produce cambios de competencias en todos los ámbitos de la empresa, desde preimpresión hasta el departamento de administración y es habitual que algunos empleados cambien de puesto. La transparencia conlleva una evaluación objetiva de la efectividad en el trabajo. Si prevalece un clima de competencia y desconfianza entre los trabajadores, rechazarán el proyecto. Es más, este rechazo puede ser provocado por el propio análisis.

Pero una vez se admite el cambio, debemos centrarnos en un tema especialmente relevante, el control de los datos digitales. Es conveniente consensuar los accesos a la información y establecer los permisos de lectura y lectura/escritura.

13.1 Implementación de Flujos de Trabajo por Módulos

La configuración de un flujo de trabajo JDF siempre resulta más fácil cuando está a cargo de un solo proveedor. Sin embargo, la imprenta debe supervisar la implementación. La adaptación de los dispositivos de la empresa y el desarrollo de los métodos de producción debe hacerse en estrecha colaboración con el proveedor para resolver los contratiempos que puedan presentarse en el momento.

El equipo administrador del WMS de la imprenta debe hacerse cargo de la gestión de la red, la integración de nuevos dispositivos, fijar los valores por defecto y tramitar los accesos de usuario. El encargado de tecnologías de la información de la empresa es el responsable del funcionamiento de los servidores que dan soporte a los diversos módulos JDF, los ordenadores, la red y control de admisión de los usuarios.

La parada de un servidor o el reinicio de un equipo por una actualización automática son imprevistos que pueden suceder durante la jornada, la empresa necesita personal capaz de resolver estas situaciones y mantener el flujo de trabajo JDF operativo. Obviamente, el administrador garantiza la estabilidad del sistema, pero si el personal implicado dispone de los conocimientos adecuados todo funcionará mejor. Desde este punto de vista, resulta imprescindible disponer de un sustituto, como mínimo, que pueda asumir estas responsabilidades (redundancia). Pero, incluso en los flujos de trabajo JDF más depurados y dependiendo del alcance de la red, la tarea de administrador puede resultar muy complicada. La tendencia habitual entre las empresas con instalaciones JDF es rodearse de expertos cualificados más que de técnicos generalistas, esto también resulta aplicable a los proveedores de soluciones JDF.

Aunque la instalación sea responsabilidad de un solo fabricante, la implementación JDF se hace por etapas y va cambiando con el paso del tiempo. Los sistemas JDF evolucionan constantemente y necesitan nuevas actualizaciones y extensiones. Antes de incorporarlas definitivamente al sistema, es importante comprobar el funcionamiento con imágenes de disco o unidades de almacenamiento alternativas, porque en caso de error es necesario recuperar al estado anterior.

Cuando en una empresa se instala un flujo de trabajo JDF, el proveedor debe revisar y comprobar la configuración con los trabajos habituales y los pedidos de clientes asiduos. Con unos productos la automatización funciona pero con otros no. Las razones que provocan esta situación pueden ser diversas, por ejemplo, si el MIS solo dispone de un modelo del catálogo o no puede acceder a una plantilla de imposición concreta, la información no llegará a producción. También se puede pasar por alto algún detalle o la descripción de un producto es defectuosa. En otros casos, la evaluación de compatibilidad entre sistemas no ha sido depurada en profundidad. Aunque suene redundante, la compatibilidad debe probarse exhaustivamente y, aún así, es posible que podamos automatizar una parte de los pedidos y otros requieran de intervención manual.

Ya comentamos anteriormente que el establecimiento de un flujo de trabajo JDF necesita reglas muy bien definidas. Pero estas normas impiden cambios en ciertos pedidos que sería más fácil resolver de otro modo. Un flujo de trabajo completamente automatizado y estrictamente regulado necesita cambios de última hora con frecuencia y dichos cambios pueden contradecir las normas adoptadas.

Si ya se ha procesado el ticket de trabajo las alteraciones suponen un esfuerzo mayor en el nuevo flujo de trabajo que con el método anterior. En los sistemas JDF complejos, los programas deben funcionar simultáneamente sobre ordenadores en red y se producen fallos, la contención de los errores no es fácil y un simple conmutador de red puede causar la caída de una aplicación, recibiendo en el MIS el mensaje de error correspondiente. Puede pasar mucho tiempo hasta que se subsana la falta y es conveniente disponer de un flujo de trabajo de emergencia que garantice el funcionamiento del sistema. Por ejemplo una prensa no recibe los preajustes y detiene la producción, una solución factible sería realizar unas planchas alternativas con los preajustes aplicados. Muchas veces las consecuencias no se ven a simple vista, por ejemplo, la prensa se detiene porque el archivo JDF no incorpora las marcas de registro. Si la comunicación JDF cesa por cualquier causa es importante que los dispositivos del flujo de trabajo puedan procesar los trabajos directamente.

En ocasiones, la oferta que nos plantea un proveedor o el sistema que planeamos resulta descabellado, la automatización tiene sus limitaciones, tanto técnicas como económicas y no tienen sentido todos los escenarios imaginables. El normal funcionamiento del proceso puede quedar paralizado por acontecimientos que necesitan la resolución del responsable de producción o la simple aprobación de un cliente.

Por ejemplo, un trabajo depende de las planchas para preparar la prensa, lo cual implica dos opciones para ajustar los parámetros de máquina, o se copian los datos JDF en la carpeta activa del MIS o el sistema del CTP envía la información directamente al sistema de producción cuando acaba de grabar las planchas. Si se interrumpe la transmisión por cualquier motivo esta configuración, tan práctica en apariencia, puede crear problemas. Las planchas y los datos de producción están disponibles, sin embargo, la prensa está parada porque el sistema no ha recibido la información.

13.2 Flujo JDF con Módulos de Múltiples Proveedores

Todos los puntos mencionados en el capítulo anterior, evidentemente, son aplicables a un flujo de trabajo JDF implementado por múltiples proveedores. Pero debemos incorporar nuevos aspectos.

En el mundo digital es bien conocido el problema del interfaz. Si el formato generado por un módulo no funciona con el programa de otro proveedor ¿quién es el responsable?

Esta cuestión es común a todos los formatos y como JMF/JDF son más recientes que PDF, seguro que surge alguna incompatibilidad. Por lo tanto, es recomendable analizar la compatibilidad exhaustivamente antes de afrontar una nueva instalación y si surge algún problema comprobar la disponibilidad de alguna actualización o extensión que resuelva la situación. Si la transmisión de datos JDF se realiza mediante carpetas activas es más fácil, pero si se utilizan paquetes MIME o mensajes JMF recibidos mediante HTTP resulta demasiado complicado interceptar y analizar la información. En algunos casos se soluciona añadiendo componentes al flujo de trabajo pero en última instancia la decisión sobre el tipo de archivo, el alcance de la automatización y el estudio de compatibilidad es única en cada empresa.

En la práctica, debemos comprobar si existe comunicación JDF real antes de completar la instalación entre dispositivos de distintos fabricantes. Si es posible analizar el interfaz directamente o solicitar un archivo JDF a un proveedor para interpretarlo y comprobar que el software del otro equipo lo importa correctamente hágalo. Habitualmente los fabricantes disponen de especificaciones para el procesamiento de archivos JDF.

Los archivos JDF se “validan” (ver sección 5.2). La página web de CIP4 alberga una utilidad en línea para validar archivos JDF, *CheckJDF* <http://jdfutility.cip4.org/JDFUtility/>. Una vez cargado el archivo nos mostrará un informe con los posibles errores, como vínculos sin resolver o atributos obligatorios inexistentes, existe una opción que marca las extensiones privadas como errores. La función *FixJDF* de la misma página, repara el archivo automáticamente, aunque con ciertas limitaciones, también permite convertir el archivo a otra versión de la especificación JDF. En la sección de descargas de CIP4, los miembros disponen de la aplicación para chequear JDF *CheckJDF.exe* con su biblioteca de enlaces dinámicos (DLL) y el programa gratuito con versiones mac y pc para editar jdf *JDF Editor* (Ver figura 13.1).

El valor *ICSVersions* indica la versión JDF utilizada. Sin embargo, nuestra experiencia demuestra que la versión no siempre corresponde a la realidad. Hay muchos errores que interrumpen el procesado de un archivo JDF, por ejemplo un recurso cumple la especificación aún estando fuera del nodo requerido cuando debería estar a nivel raíz (figura 6.5). Analizar y resolver estos problemas sin el código fuente del software importador JDF es difícil. Las primeras pruebas deben ser con archivos JDF sencillos para complicarlos paulatinamente. Tampoco debemos olvidar el objetivo, es comprobar archivos JDF con varios proveedores y para corroborar

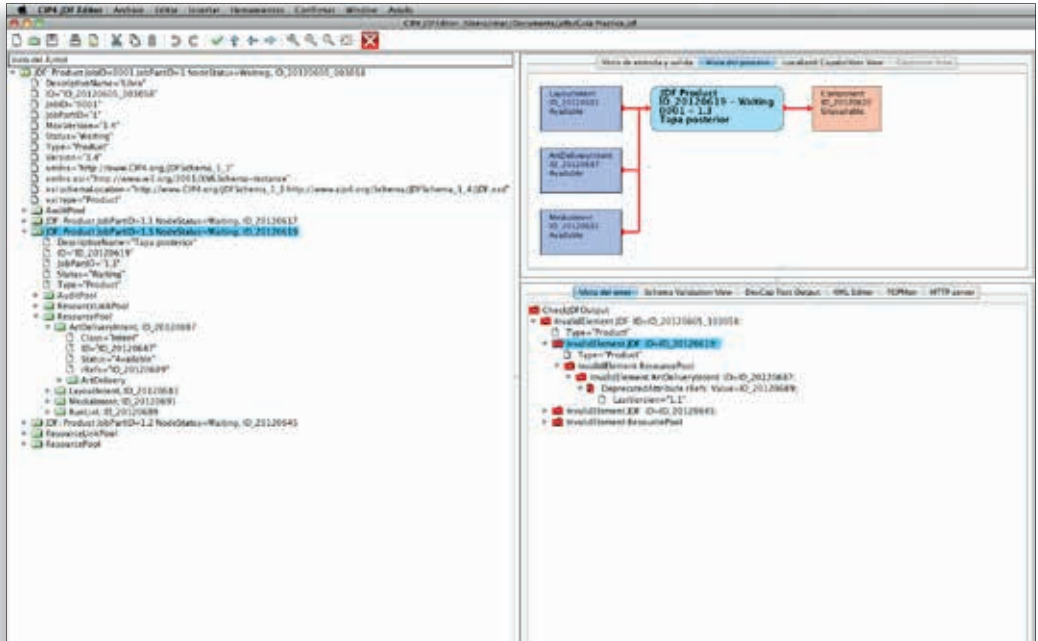


Figura 13.1
Función de validación en el Editor JDF de CIP4

la compatibilidad del software lector JDF no sirven productos simples, debemos transmitir archivos complejos que involucren a todos los dispositivos y localizar el elemento o atributo susceptible de provocar el fallo. Obviamente, ciertos errores de un proveedor no se resuelven en un tiempo razonable y debemos ser pacientes.

Los ejemplos mostrados en el libro son fragmentos seleccionados deliberadamente. Los archivos JDF reales son bastante más grandes y en determinadas circunstancias son conjuntos de folios A-4 rebosantes de nodos JDF y analizarlos puede ser bastante tedioso. Aunque JDF sea un estándar industrial, la integración de sus entornos provoca fricciones por los datos específicos del fabricante contenidos en el JDF principalmente (*Normalización* capítulo 2). La estructura JDF puede albergar datos falsos u omitir cierta información. Los interfaces JDF de los fabricantes no funcionan directamente y deben ser integrados.

Los proveedores evalúan el funcionamiento de sus soluciones con los desarrollos de los demás pero siempre surgen problemas, especialmente con los trabajos complejos. Para los usuarios sin los conocimientos adecuados, las herramientas necesarias o el tiempo para analizar el problema satisfactoriamente resultan crípticos e inexplicables.

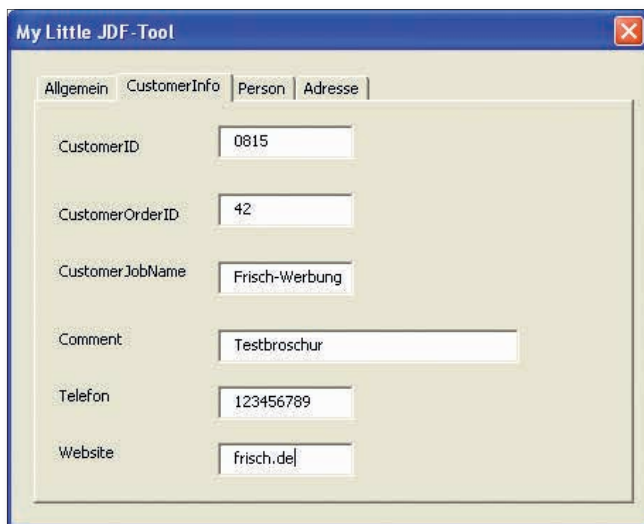


Figura 13.2
Interfaz gráfico de un
programa JDF simple

13.3 Programación JDF/JMF

En esta sección no encontrará consejos ni trucos, tampoco está dirigida a desarrolladores profesionales ni expertos en programación JDF. Está escrita principalmente, para personas con cierta experiencia en programación JAVA o C++, interesadas en iniciar un proyecto JDF.

Para programar una aplicación que importe y exporte JDF se puede usar cualquier lenguaje. El interfaz que aparece en la figura 13.2 se ha programado en Visual Basic (VBA). Si comple-

tamos los datos de los cuadros de diálogo se genera un JDF que consta de un recurso *CustomerInfo* (resultado en la figura 6.8).

La creación manual de código XML es propensa a errores y muy laboriosa, además el JDF obtenido no admite biblioteca de recursos.

En la figura 13.3 podemos ver un programa creado deprisa y corriendo que genera archivos JDF sin más pretensiones. Es más, no consigue generar ni un solo código JDF correcto, porque no tiene los atributos obligatorios tipo *Class* y *Status*. Sus consecuencias aún son más graves porque escribir software que lea e interprete estos JDF sin la biblioteca adecuada es imposible. En resumen, si nuestro objetivo es redactar software fiable estas acciones no son recomendables.

CIP4 ha creado estas bibliotecas para JAVA y C++, por lo tanto, estos son los lenguajes disponibles para la programación. Una vez decidido el entorno de programación (en este caso nos centramos en la librería de JAVA) debemos seleccionar la herramienta de una larga lista de posibilidades. Nosotros hemos elegido Eclipse [17], aunque en ocasiones también utilizamos NetBeans IDE.

El primer paso en Eclipse es definir un "Proyecto", las bibliotecas se enlazan al proyecto en forma de archivos jar. Para desarrollar nuestras propias aplicaciones JDF, necesitamos descargar de la página web de CP4, las siguientes librerías del enlace <http://www.cip4.org>:

```

Private Sub write_CustomerInfo()
    CustomerInfoID = 11101
    Print #1, " <ResourcePool>"
    Print #1, " <CustomerInfo CustomerID=" & "" & CustomerID & "" & "
        CustomerJobName=" & "" & CustomerJobName & ""
    Print #1, " CustomerInfoID=" & "" & CustomerInfoID & "" & "
        CustomerOrderID = " & "" & CustomerOrderID & "" & " > "
    Print #1, " <Comment>" & Comment & "</Comment>"
    Print #1, " <Contact Class=" & "" & "Parameter" & "" & "
        ContactTypes=" & "" & "Customers" & ""
    Print #1, " ID=" & "" & ContactRefl & "" & "
        Status=" & "" & "Available" & "" & ">"
    Print #1, "<Person FirstName=" & "" & FirstName & "" & "
        FamilyName=" & "" & FamilyName & "" & "
        NamePrefix=" & "" & NamePrefix & "" & ">"
    Print #1, " <ComChannel ChannelType=" & "" & "Phone" & "" & "
        Locator=" & "" & "Telefon" & "" & "
        Status=" & "" & "Available" & "" & "/>"
    Print #1, " <ComChannel ChannelType=" & "" & "WWW" & "" & "
        Locator=" & "" & "Website" & "" & "
        Status=" & "" & "Available" & "" & "/>"
    Print #1, "</Person>"
    Print #1, "<Address City=" & "" & City & "" & "
        Street=" & "" & Street & "" & " Country=" & "" & Country & "" & "
        PostalCode=" & "" & PostalCode & "" & "/>"
    Print #1, " </Contact>"
    Print #1, " </CustomerInfo>"
    Print #1, " </ResourcePool>"
End Sub

```

- XercesImpl.jar (Version 2.9.0).
- Commons-lang.jar (Version: 2.3).
- JDFLib.jar (Version 2.1.3.4).

Figura 13.3
Aplicación compleja
programada en Visual Basic
para crear documentos JDF

Hemos programado una aplicación muy sencilla con las siguientes características (ver figura 13.4):

- Lectura y visualización de archivos JDF procedentes de una carpeta activa seleccionada.
- Al hacer doble clic sobre un trabajo se muestran detalles del mismo. (ID de trabajo, nombre de la orden, tamaño y tirada).
- Definición de 4 botones de comando en el interfaz: "inicio", "pliegos desechados", "pliegos correctos" y "fin de trabajo".
- Se instala un sensor fotosensible en una prensa sin interfaz JDF. Para medir, el empleado selecciona la opción correcta "pliegos OK" o "pliegos KO" y presiona el botón de "inicio".

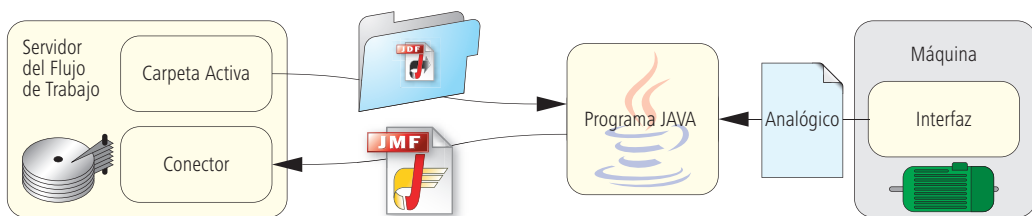


Figura 13.4
Proyecto para conectar una prensa analógica al flujo de trabajo mediante JDF/JMF

- Al presionar el botón “fin de trabajo”, el programa envía un mensaje JMF a la dirección seleccionada (puerto TCP e IP).

No podemos mostrar todo el código completo, solo aparecen los fragmentos más interesantes.

El subproceso *Listar Ventana (ListWin)* de la figura 13.5, activa la vigilancia de la carpeta activa a intervalos regulares. Si detecta actividad elimina las entradas anteriores e importa los nuevos trabajos mostrándolos en la ventana principal.

Figura 13.5
Creación de carpeta activa

```
public Main window () { // Constructor
    ...
    ListWin = new Thread(this);
    ListWin.start();
    ...
}
public void run() {
    while (true) {
        try {
            jobliste.removeAll();
            dateien = ordner.list();
            for (int i =0; i< file.length; i++){
                jobliste.add(dateien[i]);
            }

            this.add(jobliste);
            repaint();
            Thread.sleep(10000);
        }
        catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```


La figura 13.6 primero *Valida (Parsing)* el *Archivo JDF (JDFDoc)* y luego identifica el *Elemento Raíz* con el nombre *Root*. A continuación mostramos el código que extrae los atributos del elemento raíz. `String JobPartID = Root.getJobID(true)`. Otros elementos de *Root* que nos permiten localizar y obtener recursos son *CustomerInfo* y *ResourcePool*, también denominado *RSP*, como podemos ver es bastante simple. Pero hemos dejado un par de cuestiones de seguridad para el final, hemos asumido sin más, que el recurso *CustomerInfo* existe y que este, a su vez, contiene el atributo *CustomerJobName*.

```
JDFDoc JDFDocument = JDFDoc.parseFile(pfad + dateiname);
final JDFNode Root = JDFDocument.getJDFRoot();
...
String JobPartID = Root.getJobID(true);
...
JDFResourcePool RSP = Root.getResourcePool();
final JDFResource CustomerInfo = RSP.getResource("CustomerInfo", 0, "");
CustomerJobName = CustomerInfo.getAttribute("CustomerJobName");
```

Lo último que presentamos es el código fuente de la figura 13.7 para enviar señales JMF. La cabecera HTTP y el mensaje JMF van ensamblados. Se crea el nodo *Identificativo del Dispositivo (DeviceInfo)* y el elemento de nodo *Fase del Trabajo (JobPhase)*, a continuación se rellenan los atributos y se escribe la secuencia (*String Content*) y tras crear la conexión con la dirección IP y el puerto se redacta el mensaje JMF y la cabecera HTTP. El resultado de este pequeño programa es una señal JMF del tipo que aparece en la figura 13.8.

En ocasiones, programar una aplicación JDF/JMF completa es excesivo y modificar un archivo JDF es más que suficiente o quizás nos sirva transformar el interfaz XML específico de un proveedor. El **Lenguaje Extensible de Transformación de Hojas de Estilo XSLT** es la herramienta que permite convertir un archivo XML. Los archivos **XSLT** controlan la transformación en otras formas de XML. Pero existen muchos libros [9] que nos pueden ayudar y no profundizaremos en este tema.

Figura 13.6
Lectura y análisis de
atributos

```

JMFMessage(String JobID,String JobPartID, Long goodCount){
    String DeviceStatus = "Running";
    //Construcción de un mensaje JMF, es decir,el contenido del paquete http-
    JDFJMF JMF = JDFJMF.createJMF(EnumFamily.Signal,EnumType.Status);
    KElement DI = JMF.appendElement("DeviceInfo");
    DI.setAttribute("DeviceStatus", DeviceStatus);
    KElement JP = DI.appendElement("JobPhase");
    JP.setAttribute("TotalAmout", goodCount.toString());
    JP.setAttribute("JobID", JobID);
    JP.setAttribute("JobPartID", JobPartID);
    String content = JMF.toXML();

    //Construcción de la cabecera http
    long length = content.length();
    String header = "HTTP/1.1 200 OK" + "\nContent-Length: " + length +
        "\nContent-Type: text/html";

    ...
}

```

```

HTTP/1.1 200 OK
Content-Length: 509
Content-Type: text/html

<?xml version="1.0" encoding="UTF-8"?>
<JMF xmlns="http://www.CIP4.org/JDFSchem_1_1"
    timeStamp="2008-12-02T18:32:24+01:00" Version="1.3">
  <!--generado por el código abierto JAVA CIP4 Librería JDF versión:CIP4 JDF
  Writer Java 1.3 BLD 40-->
  <Signal ID="m081202_063224609_000000" Type="Status"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="SignalStatus" />
    <DeviceInfo DeviceStatus="Running">
      <JobPhase JobID="_08-0157" JobPartID="_08-0157" TotalAmout="300" />
    </DeviceInfo>
  </JMF>

```

Figura 13.7 (Superior)
Código fuente para el envío
del mensaje JMF.

Figura 13.8 (Inferior)
Mensaje JMF con la
cabecera HTTP

Bibliografía

(N.d.T.: Publicaciones en inglés excepto las señaladas).

- [1] Adobe System:
Libro Blanco de Adobe PDF Print Engine 2
<http://www.adobe.com/products/pdfprintengine/> (2008)
- [2] Adobe System Incorporated:
Especificación XMP
<http://www.adobe.com/devnet/xmp/> (2004)
- [3] Adobe System Incorporated:
Revisión TIFF 6.0,
<http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf> (1992)
- [4] Adobe System Incorporated:
Referencia de Lenguaje PostScript, 3. Ausgabe,
Editorial Addison Wesley,
ISBN 0-201-37922-8 (1999) o
<http://www.adobe.com/products/postscript/pdfs/PLRM.pdf>
- [5] Adobe System Incorporated:
Referencia PDF 6ª edición, Adobe Portable
Document Format, Version 1.7,
http://www.adobe.com/devnet/pdf/pdf_reference_archive.html (2006)
- [6] Adobe System Incorporated:
Formato Portable Job Ticket, 1.1,
Notas Técnicas # 5620 (1999)
- [7] ANSI:
IT8.6-2002 (R2007) Tecnología Gráfica–
Intercambio de Datos Digitales en Preimpresión
– Datos de Troquelado (Diecutting) (DDES3)
- [8] Apple Inc.:
Resumen AppleScript,
<http://developer.apple.com/documentation/AppleScript/Conceptual/AppleScriptX/AppleScriptX.html> (2007)
- [9] Behme, Henning und Mintert, Stefan:
XML en la Práctica: Publicación Web
Profesional con Lenguaje XML (Alemán),
Addison Wesley, ISBN 3-8273-1636-7 (2000)
- [10] Bohan, Mark y otros:
Automatización y Flujo de Trabajo JDF,
2006 TAGA Proceedings
- [11] CGATS.20:
Intercambio de Datos Variables con PPML y PDF
(PPML/VDX),
www.npes.org (2002)
- [12] CIP4:
Especificaciones de Conformidad e
interoperabilidad, Version 1.3
www.cip4.org/ (2007/2008)
- [13] CIP4:
Especificación JDF 1.4,
www.cip4.org (2008)
- [14] CIP4:
El Mercado JDF;
<http://www.cip4.org/marketplace/>
- [15] Manual de XML Comercial (cXML),
Versión 1.2.019, <http://www.cxm.xml.org> (2008)
- [16] Dolin, Penny Ann:
Explorando el Flujo de Trabajo Digital,
Thompson Delmar Learning,
ISBN 1-4018-9654-5 (2006)
- [17] Fundación Eclipse:
Eclipse IDE para Desarrolladores Java,
<http://www.eclipse.org/downloads>
- [18] Freund, Jacob; Götzer, Klaus:
Prácticas Productivas con Flujos de Trabajo,
Editorial Carls Hanser Munich (Alemán),
ISBN 978-3-446-41482-2 (2008)
- [19] Grupo de Trabajo Ghent PDF:
PDF/X Plus
www.gwg.org
- [20] Hamilton, Eric:
Formato de Intercambio JPEG, Version 1.02,
<http://www.jpeg.org/public/jfif.pdf> (1992)
- [21] Harvey, James:
Grupo de Usuarios de JDF
<http://www.printing.org/jdfusergroup>
- [22] Hoffmann-Walbeck, Thomas:
Producción de Planchas Digitales (Alemán),
Editorial dpunkt, ISBN 3-89864-182-1 (2004)
- [23] Organización Internacional de Estandarización
(ISO) 15930: PDF/X Partes 1 a 8,
Editorial Beuth (2001 – 2008)

- [24] Organización Internacional de Estandarización (ISO) 12647: Partes 1 a 7, Editorial Beuth (2001 – 2007)
- [25] Organización Internacional de Estandarización (ISO) 16612-2: Tecnología Gráfica – Intercambio de datos Variables Parte 2: En uso PDF/X-4 y PDF/X-5 (PDF/VT-1 y PDF/VT-2) en desarrollo (2008)
- [26] Consejo Internacional de Telecomunicación y Prensa (IPTC): <http://www.iptc.org>
- [27] Asociación Japonesa de Industrias Electrónicas y de Tecnologías de la Información (JEITA): <http://www.jeita.or.jp/english/>
- [28] JEITA: Formato de Intercambio de Imágenes para Cámaras Digitales: Exif Versión 2.2, JEITA CP-3451 (2002)
- [29] Kipphan, Helmut: Manual de Medios Impresos (Alemán), Editorial Springer Berlin, Heidelberg, New York, ISBN 3-540-66941-8 (2000)
- [30] Kodak Graphic Communications: "Automatización Basada en Reglas con el Software Prinergy 4," Libro Blanco, <http://graphics1.kodak.com/documents/Release%20date.doc> (2007)
- [31] Koster, Kai: Tecnología de la Información y Comunicación para Empresas (Alemán), Editorial Carl Hanser, ISBN: 3-446-2118-3 (1999)
- [32] Kühn, Wolfgang; Grell, Martin: JDF: Integración de Proyecto, Tecnología y Descripción de producto (Alemán), Editorial Springer Berlin Heidelberg, ISBN 3-540-20893-3 (2004)
- [33] Kular, Christopher: Formato de Definición de Trabajo y los Medios Impresos, International Journal Of The Book, ISBN: 1447-9516 (2007)
- [34] Microsoft: Windows Script Host, <http://www.microsoft.com/downloads>
- [35] Mittelhaus, Michael: Como Automatizar una Imprenta, (Alemán) Bundesverband Druck und Medien e.V (bvdm), Art-Nr. 83107 (2005)
- [36] PODi - the Digital Printing Initiative: Personalized Print Markup Language – Especificación Funcional, Versión 2.2 (2006)
- [37] PODi - the Digital Printing Initiative: Personalized Print Markup Language – Especificación Imposición, Versión 2.2 (2006)
- [38] PODi - the Digital Printing Initiative: Personalized Print Markup Language – Especificación Convenciones Arte, Versión 2.2 (2006)
- [39] PODi - the Digital Printing Initiative: Ticket Digital de Imprenta – Impresión con PPML y JDF, Versión 2.0 (2005)
- [40] PrintTalk Version 1.3, <http://www.cip4.org> (2007)
- [41] Schwabe, Gerhard y otros (Hrsg.): CSCW-Konpendium (Alemán), Editorial Springer, ISBN 3-540-67552-3 (2001)
- [42] St. Laurent, Simon; Fitzgerald, Michael: XML, Editorial O'Reilley Köln, ISBN: 13 978-3-89721-516-0 (2006)
- [43] Workflow Management Coalition, Consorcio para la descripción de WMS www.wfmc.org
- [44] W3C (Dave Beckett): Marco para la Descripción de Recursos (RDF), <http://www.w3.org/RDF/>
- [45] W3C: Lenguaje Descripción de Servicios Web (WSDL) Versión 2.0, <http://www.w3.org/TR/wsdl20-primer/> (2007)

Glosario

Acabado en línea	Operaciones de fabricación tales como la numeración, direccionamiento, alzado, plegado, troquelado y conversión que se realizan como parte de una operación continua tras la impresión en una prensa o en un solo dispositivo como parte del proceso de encuadernado. Es habitual contar con acabado en línea en operaciones de impresión rotativa.
Ajuste	Preparación de una máquina o dispositivo.
ASCII85	Conjunto de caracteres ASCII imprimibles representados con codificación de transferencia de 85 patrones de bits diferenciados.
Bitmap	Imagen representada por una matriz de elementos básicos de imagen o píxeles codificados como uno o más dígitos binarios.
Bytemap	Datos de imagen estructurados en 8 bits y 256 niveles por color como mínimo.
Carpeta activa	Interfaz unidireccional entre dos módulos de software asociados a una carpeta. Un módulo de software escribe datos en la carpeta regularmente mientras el otro módulo controla la llegada de archivos.
Compresión RLE	En inglés "Run-length compression". Modelo de compresión sin pérdida de datos en la que se almacena el número de repeticiones como un único valor más el recuento.
Curvas de transferencia	Tabla de valores o curvas utilizadas para alterar los valores tonales de un RIP también denominadas "curvas de compensación" o "curvas de calibración".
Dirección del grano	Dirección de las fibras (grano) del papel.
Folio alto/bajo	Grapado a caballete. Excedente de papel (en anverso o reverso) para la apertura automática de las hoja plegadas utilizado para el encuadernado a caballete con grapas.
Frecuencia de trama	Trama de semitonos cuya medida es equivalente a la cantidad de líneas o puntos por pulgada (Lpi). Cuanto mayor es la densidad de trama mayor es el detalle en el semitono. La resolución de trama utilizada varía entre las bajas lineaturas utilizadas en cartelería de gran formato hasta las 200 lpi o más utilizadas en la impresión de alta calidad sobre papel estucado.
Gestión de color	La gestión de color la realizan técnicos experimentados con dispositivos científicos y programas especializados. La aplican los usuarios mediante motores de gestión de color y programas adaptados determinando automáticamente las características de reproducción óptima de escáneres, monitores y dispositivos de salida.

Imposición	Organización en la etapa de preimpresión de las páginas del trabajo sobre la forma de impresión para que el contenido este correctamente orientado y dispuesto para su posterior impresión, plegado, guillotinado y encuadrado insertando todas las marcas necesarias para la fase de prensa y acabado. La maquetación de página es la definición de los elementos constituyentes del diseño, como cabeceras, textos o numeración de páginas, mientras que imposición es la disposición de las páginas completas sobre uno o más pliegos de impresión mayores que las propias páginas. El tipo de encuadrado afecta al orden del montaje y la imposición. En un flujo de trabajo digital de impresión se utiliza un software especial para imposición.
Impresión comercial	Impresión destinada a negocios y uso privado como tarjetas de visita, papelería empresarial, folletos y catálogos promocionales. Generalmente no se incluyen libros y embalaje o publicaciones periódicas como revistas o periódicos.
Interpretación	Transformación que se produce desde un lenguaje de descripción de página al formato interno de datos de un RIP también denominado "DisplayList".
Margen de pinza	Distancia desde el borde inferior de la plancha hasta el borde inferior del papel en impresión offset de pliegos. Área utilizada por las pinzas para extraer el papel de la prensa. Este margen no puede contener imágenes.
Montaje	Composición de imágenes en película sobre un substrato transparente para transferirla por fotoexposición a la plancha para su impresión posterior.
Mosaico	(En inglés tiling). Repetición del mismo producto sobre un pliego de impresión, como páginas o cajas de cartón plegables desplegadas.
Normalización	Conversión de datos de contenido específicos del cliente a formato JDF estandarizado.
Plantilla de Imposición	Documento utilizado como base para realizar una imposición basada en una secuencia de plegado en particular, conteniendo las signaturas, la numeración de páginas, guías de límite, márgenes de pinza y marcas necesarias.
Preajuste zonas de tinta	Ajuste previo del entintado de una prensa offset basado en los datos de un trabajo realizable por las zonas en que se dividen los cilindros.
Preflight	Verificación previa mediante una función o software específico de todos los componentes de un trabajo (textos e imágenes en formato digital) para identificar problemas potenciales que provoquen la repetición o corrección del trabajo y enviar el documento correcto para su salida en alta resolución.
Profundidad de color	Número de bits necesarios para almacenar un píxel o tono.
Prueba ferro	También denominada prueba de contenido. Impresión sin precisión cromática realizada habitualmente en impresoras de chorro de tinta, destinada a la comprobación de marcas de control y contenidos antes de pasar a prensa.
Rasterizado	(En inglés Rendering). Conversión de la descripción matemática de objetos compuestos en lenguaje específico de páginas (por ejemplo, PDF) a imágenes de medio tono.

RIP	(Raster image processor). Procesador de imágenes rasterizadas, software gestionado por un lenguaje de descripción de página, por ejemplo PostScript, específico para la interpretación de imágenes rasterizadas determinando el valor de cada píxel y convirtiéndolo al punto de salida correspondiente. El mapa de bits resultante se utiliza para guiar el láser de una filmadora o CTP y graba la imagen a la película o plancha.
Sistemas gestor de pedidos	(MIS - Management Information System). Software para administrar pedidos, gestionar datos de producción y realizar el cálculo integrado de los costes.
Sobrante	(En inglés Creep o Shingling). Compensación de la distancia de las hojas de un libro al lomo provocada por el grosor del papel.
TIFF-B	Formato de archivos para el almacenamiento de mapas de bits en binario.
Tipo de trama	Los tipos de trama utilizados en artes gráficas se dividen en AM y FM. Trama convencional y trama estocástica respectivamente.
Tirada	Impresión de los pliegos correctos de un trabajo tras el ajuste de la prensa.
Tramado	Proceso de conversión de una imagen de tono continuo en una matriz de puntos cuyo tamaño varía para simular la apariencia de una imagen tonal.
Trama AM/FM	AM, acrónimo de “modulación ampliada,” es un tipo de grabado de planchas basado en tramas de semitonos con puntos que varían en tamaño y al ser impresos causan la percepción de áreas de altas luces a sombras profundas. FM significa “frecuencia modulada” y es una trama de semitonos donde los puntos son sustituidos por filamentos, los tonos se crean por concentración. (Las áreas se diferencian por baja concentración en luces y alta en sombras).
Trapping	Reventado. (1) Proceso de impresión de una tinta húmeda sobre una tinta previamente impresa seca o húmeda. (2) Creación en preimpresión de pequeñas superposiciones entre colores adyacentes para compensar los errores de registro que se producen en la prensa y provocan la aparición de huecos entre los colores.
TrimBox	Caja de corte o casilla de límite de página. Definición de las dimensiones finales de la página terminada después del recorte, puede ser inferior que la caja de medios o MediaBox, necesaria para la inclusión de instrucciones de impresión o marcas de control de color o registro entre otras. En documentos PDF representa el tamaño de la página.
Web-to-print	Solución de software basado en internet. Mediante un interfaz de usuario soportado por el navegador este puede crear, solicitar y aprobar la impresión de un producto a un impresor. En la fase de creación permite el diseño de un producto impreso en línea utilizando como base plantillas estandarizadas.

Abreviaturas

B2B	Business-To-Business Transacciones Comerciales Electrónicas utilizando la tecnología Electronic Data Interchange.
CFF2	Common File Format, Version 2 Estándar Industrial CFF2 versión 2, utilizado para el intercambio de diseños entre CAD y Packaging.
CIE	Commission Internationale de l'Éclairage Comisión Internacional de la Iluminación fundada en 1931 y con sede en Viena, Austria.
CIM	Computer Integrated Manufacturing Manufactura Integrada por Computador, concepto acuñado por Harrington J. Jr 1973
CIP3	International Cooperation for Integration of Prepress, Press and Postpress Cooperación Internacional para la Integración de Preimpresión, Impresión y Acabado.
CIP4	International Cooperation for Integration of Processes in Prepress, Press and Postpress Cooperación Internacional para la Integración de Preimpresión, Impresión y Acabado.
CSCW	Computer Supported Cooperation Work Trabajo Cooperativo mediante Soporte Informático.
CTM	Current Transformation Matrix Matriz de Transformación en Uso.
CtP	Computer-to-Plate Directo a Plancha.
cXML	Commerce Extensible Markup Language Lenguaje de Marcado Extensible para Comercio Electrónico.
DDES	Digital Data Exchange System Sistema para el Intercambio de Datos Digitales.
dpi	Dots per Inch Puntos por Pulgada.
DLL	Dynamic Link Library Librería de Enlaces Dinámicos.
DTD	Document Type Definition Definición de Tipo de Datos en XML.
DTP	Desktop Publishing Publicación de Escritorio o Autoedición.
DXF	Drawing Exchange Format Formato Informático de Dibujo Computarizado, utilizado principalmente por el programa AutoCAD.
EAN	European Article Number Sistema de Códigos de Barras.
ECMA	European Carton Maker Association Asociación Europea de Fabricantes de Cartón.

EPS	Encapsulated PostScript PostScript encapsulado, o EPS, formato de archivo gráfico.
ERP	Enterprise Resource Planning Sistemas de Planificación de Recursos Empresariales.
EXIF	Exchangeable Image File Format Formato de Imagen para Cámaras Digitales creado por la Asociación Industrial Japonesa (JEITA).
FEFCO	Fédération Européenne des Fabricants de Carton Ondulé Federación Europea de Fabricantes de Cartón Ondulado.
GPS	Global Positioning System Sistema de Posicionamiento Global.
HdM	Hochschule der Medien (Stuttgart Media University) Universidad de Ciencias Aplicadas Hochschule der Medien (Stuttgart)
HTML	Hypertext Markup Language lenguaje de Marcado de Hipertexto predominante en la elaboración de páginas web.
HTTP	Hypertext Transfer Protocol Protocolo de Transferencia de Hipertexto.
HTTPS	Hypertext Transfer Protocol Secure Protocolo Seguro de Transferencia de Hipertexto utilizado principalmente para enviar datos personales.
ICC	International Color Consortium Consortio Internacional del Color fundado en 1993 para conseguir una gestión común del Color.
ICS	Interoperability Conformance Specification Especificación de Conformidad e Interoperabilidad.
IPTC	International Press Telecommunication Council Comité Internacional de Telecomunicaciones de Prensa fundada en 1965 y con sede en Windsor (UK).
JDF	Job Definition Format Formato de Definición de Trabajo.
JFIF	JPEG File Interchange Format Formato de Intercambio de Archivos JPEG.
JPEG	Joint Photographic Experts Group Grupo Mixto de Expertos en Fotografía/Formato Digital de Imagen con función de compresión variable.
JEITA	Japan Electronics and Information Technology Industries Association Asociación Industrial Japonesa de Tecnologías de la Información y la Electrónica.
JMF	Job Messaging Format Formato de Mensajes de Trabajo.
JTP	Job Ticket Processor Procesador de Tickets o Partes de Trabajo Electrónicos.
MIME	Multipurpose Internet Mail Extensions Extensiones Multifunción de Correo dirigidas al intercambio de archivos por Internet (texto, audio, vídeo).
MIS	Management Information System Sistemas de Gestión de la Información.

OMS	Order Management System Sistemas de Gestión de Pedidos.
PDF	Portable Document Format Formato de Documento Portátil de Adobe en Lenguaje de Descripción de Página.
PDF/VT	PDF Variable Transactional PDF Transaccional para Datos Variables.
PDL	Page Description Language Lenguaje de Descripción de Página.
PJTF	Portable Job Ticket Format Formato de Parte de Trabajo Portátil.
PPF	Print Production Format Formato de Producción de Impresión.
ppi	Pixel per Inch Píxeles por Pulgada.
PPML	Personalized Print Markup Language Lenguaje de Marcado para Impresión Personalizada.
PS	PostScript Lenguaje de Descripción de Objetos original de Adobe.
RBA	Rules-Based Automation Automatización basada en Normas de Kodak.
RDF	Resource Description Framework Marco de Descripción de Recursos.
RFQ	Request for Quote Solicitud de Presupuesto.
RIP	Raster Image Processor Procesador de Imágenes Rasterizadas.
SOAP	Simple Object Access Protocol Protocolo de Acceso a Objetos Simples.
TIFF	Tag(ged) Image File Format Formato Etiquetado de Archivo de Imagen.
UCS	Universal Multiple-Octet Coded Character Set Codificación Universal de Caracteres basado en Octetos.
URI	Uniform (Universal) Resource Identifier Identificador Universal de Recursos.
URL	Uniform Resource Locator Localización Universal de Recursos.
UTF	UCS Transformation Format Formato de Transformación de Caracteres Unicode.
VDP	Variable-data Printing Impresión de Datos Variables.

W3C	World Wide Web Consortium	Consortio Internacional para la Normalización del Uso y Crecimiento de la Web mediante W3C.
WfMC	Workflow Management Coalition	Consortio para la Interoperabilidad de Flujos de Trabajo, responsable de las normas XPDL y Wf-XML.
WMS	Workflow Management System	Sistema Gestor de Flujos de Trabajo.
XSD	XML Schema Definition	Esquema para describir la estructura los contenidos y restricciones de los documentos XML.
XML	Extensible Markup Language	Lenguaje de Marcas Extensible.
XMP	Extensible Metadata Platform	Plataforma Extensible para Metadatos.
XPS	XML Paper Specification	Lenguaje Abierto XML de Descripción de Páginas.
XSLT	Extensible Stylesheet Language Transformation	Lenguaje Extensible de Transformación de Hojas de Estilo para documentos XML.

Índice

A

Acabado en Línea de Rotativa
 (WebInlineFinishing) 146, 207

Acabado (Finishing) 7, 2, 5, 6, 13,
 27, 32, 36, 37, 45, 51, 78,
 106, 146, 160, 161, 167, 169,
 179, 181, 182, 194, 207, 208

Aceptado (Available) 141

Acuse de recibo (Acknowledge) 96

Adaptabilidad 20, 24

Adhesivo de Montaje
 (MountingTape) 148

Administrador (Manager)
 84, 85, 86, 92, 117

AFCO 182

Agente 72, 79, 83, 84,
 125, 145, 173

Agente (Agent) 79

Ajuste 97

Ajuste de grabado (PlateSetting) 128

Ajuste de Plancha
 (PlateSetting) 127, 137

Ajustes de Imagen (ImageSetting) 141

Analizador XML (Parser) 59, 71

Ancho de reventado (TrapWidth) 137

Anchura de Negro (BlackWidth) 137

Ansi DDES3 183

Anverso (Front) 74

Aprobación (Approval) 141

Aprobación Impresión Digital
 (DigitalPrinting Approval), 141

Aprobación Realizada
 (ApprovalSuccess) 141

Arquitectura Extreme RIP 39, 50

Ascendiente (Ancestor) 82

ASCII85 207

Asignación de valor 55

Atributo 55, 56, 57, 59, 61, 66,
 67, 68, 69, 71, 73, 76, 82, 86,
 93, 97, 109, 113, 114, 115,
 118, 121, 128, 133, 136, 137,
 141, 143, 149, 151, 152, 154,

155, 165, 170, 173, 175, 179,
 182, 185, 188, 198, 199, 200

Atributo separado
 (rRefsROCopied) 82

Auditoría (Audit) 52, 72, 92

Automatizar por Normas (RBA)
 125, 126

Autor (Author) 72, 115

B

Barnizado (Varnishing) 145

Base de datos (Database) 6, 12, 78,
 92, 103, 163, 165, 166, 191

Base ICS 84, 85, 86

BillingCode 115

Bitmap 207

Bytemap 207

C

CAD 182

Caja de Corte (TrimBox) 135, 209

Caja de límites (BoundingBox)
 133, 182

Caja Gris (GrayBox) 74, 75,
 107, 111, 112, 126, 127,
 128, 130, 137, 173, 177

Caja Relativa (RelativeBox)
 108, 109, 184

Cálculo 103

Cálculo/costes 101

Cálculo zonas de entintado (
 InkZoneCalculationParams)
 150, 159

Cálculo zonas entintado
 (InkZoneCalculation) 75

Calibración Tono Continuo
 (ContoneCalibration) 138

Cámaras digitales 39

Camisa de flexo (Sleeve) 179

Canal de Comunicación
 (ComChannel) 56, 117

Canal de comunicación detenido
 (StopPersistentChannel) 145

Canal del cartón ondulado (Flute) 179

Cancelación (Cancellation) 121

Cantidad (Amount) 69, 70,
 121, 158, 162

Capacidad de Dispositivos
 (DeviceCapabilities) 173

Cara (Side) 74, 113, 134, 149,
 152, 154, 155, 157

Carpetas activas (hotfolders)
 89, 90, 91, 92, 95, 98,
 125, 126, 145, 198, 207

Cartón ondulado
 (CorrugatedBoard) 179

Catálogo de plegado
 (FoldCatalog) 109

Cauchos (Blanking Tool, Blanker) 181

CheckJDF 198, 222

Cinta adhesiva de montaje
 (MountingTape) 148, 179

CIP3 1, 24, 97, 210, 222

CIP4 4, 1, 6, 7, 36, 57, 58, 67,
 68, 78, 87, 118, 122, 146,
 159, 167, 195, 198, 199,
 200, 204, 205, 210, 222

Clase (Class) 68, 69, 71, 73, 74,
 86, 107, 108, 109, 113, 115,
 137, 140, 142, 149, 151,
 152, 154, 155, 156, 157,
 158, 170, 172, 176, 183, 184,
 187, 188, 190, 200, 201

Clase del papel (Grade) 155

Claves de partición (PartIDKeys) 73,
 74, 108, 109, 113, 134, 149, 152,
 155, 156, 157, 158, 187, 188

ClipBox 134

Cobertura zona X (ZoneSettingX) 152

Cobertura zona Y (ZoneSettingY) 152

Códigos de barras 167, 179,
 189, 190, 191

ColorIntent 111

ColorPool 156

ColorType 156

Comando de recurso
 (ResourceCommand) 95

Comando JMF 92

Combinado 77, 81

- Compensación del trazo 179
- Componente 69, 70, 73, 106, 111, 148, 161, 169, 175, 178, 185, 186
- ComponentLink 70, 113, 158, 162, 187
- ComponentType 69, 73, 158
- Compresión RLE 207
- Comunicación constante (PersistentChannel) 145
- Condición (Condition) 73, 158
- Condiciones de impresión (PressConditions) 147
- Condiciones de Impresión (PrintCondition) 148, 149
- Confirmación del pedido (Confirmation) 121
- Consorcio para la descripción de WMS (WfMC) 206
- Consulta JMF 92, 93
- Consulta (OrderStatusRequest) 92, 93, 96, 98, 121
- Contacto (contact) 56, 115, 141
- Contenido por Cara (SurfaceContentsBox) 133, 134, 135, 151, 154
- Contra matriz (CounterDie) 181
- Controlador (Controller) 79
- Control de Color (ColorantControl) 76, 113, 132, 136, 157
- Control de Transferencia (TransferFunctionControl) 138
- Conversión al canal negro (RGBGray2Black) 143
- Conversión de Formato 138, 140
- Conversión de gris RGB a Negro (RGBGray2Black) 143
- Conversión Espacio de Color (ColorSpaceConversion) 142
- Corte (Cutting) 169
- Corte de Bloque (CutBlock) 186
- Corte de Forma (ShapeCutting) 185, 215
- Creación de Contenido (ContentCreation) 127
- Creación del Troquel (DieLayoutProduction) 183
- Creación del Troquel (DieMaking) 183
- Creación de página (LayoutElementProduction) 164
- CSCW 13, 206, 210
- Curvas de calibración 48
- Curvas de compensación de ganancia de punto 48
- Curvas de incremento 48
- Curvas de linearización 48
- Curvas de transferencia 48, 207
- cXML 55, 61, 62, 118, 120, 205, 210
- D**
- Datos de Contenido (Contentdata) 15, 16, 17, 18, 19, 39, 83, 102, 107, 111, 125, 126, 157, 184, 208
- Datos de Trapping (TrappingDetails) 136
- Datos variables (VDP) 163, 165, 212
- Declaración 55
- Definición de Forma (ShapeDef) 183
- Denegado (Refusal) 121
- Densidad Máxima de Negro (BlackDensityLimit) 136
- Densidad neutra de color (NeutralDensity) 156
- Desconocido 97
- Descripción de objeto 15
- Detalles de MIS (MISDetails) 96
- Detenido (Stopped) 68, 149
- Devolución del trabajo (ReturnJob) 121
- Diagrama de actividad 25, 30, 180
- Diagrama de flujo 25, 26, 36, 37, 38, 168
- Diagrama de transición 25, 27
- Dimensión del SoporteX/Y (BlankDimensionsX/Y) 188
- Dimensión (Dimension) 71
- Dimensiones del Contenido (SurfaceContentsBox) 133, 134, 135, 151, 154
- Dirección del grano (GrainDirection) 155, 207
- Diseño estructural 179
- Disponible (Available) 69
- Dispositivo (Device) 79, 111, 175
- División de Recursos 72
- Doble (Crease) 188
- DropIntent 69
- DTD 58, 210
- DTP 71, 133, 173
- Dublin Core Schema 43
- E**
- ECMA 182
- Elaboración de Pruebas/Planchas (ProofAndPlateMaking) 127
- Elemento raíz 45, 46, 52, 56, 60, 61, 63, 65, 67, 69, 70, 74, 86, 91, 111, 122, 203
- Elementos 19, 32, 33, 51, 55, 56, 58, 59, 60, 62, 67, 71, 83, 92, 97, 109, 114, 115, 118, 120, 121, 131, 133, 136, 142, 143, 145, 147, 153, 163, 165, 170, 173, 185, 203, 207, 208
- Elementos de Maqueta (LayoutElement) 165
- El primero que llega primero sale (First In/First Out) 93
- Empresa (Company) 56
- Encuadernación (Binding) 29, 103, 175, 208
- En Curso (InProgress) 68
- Enlace de medios (MediaLink) 139
- Enlaces entre Recursos (ResourceLinks) 111
- Ensamblado (Assembly) 110, 175, 176
- Entrada en Cola (SubmitQueueEntry) 98
- Envase flexible metalizado (Foil) 179
- Envases flexibles metalizados 179
- Espacio de Color de Origen 143
- Espacio de nombres XML 57

E

Especificación de Archivo
 (FileSpec) 143

Especificaciones Conformidad de Interoperabilidad (ICS) 84, 98

Especificación JDF 3, 6, 7, 23, 55, 63, 68, 114, 173, 175, 178, 182, 188, 198

Esperando Aprobación
 (WaitingForApproval) 149, 215

EsquemaDB (DBSchema) 165

Esquema JDF (JDF.xsd) 58

Esquema XML 58

Estado del dispositivo
 (DeviceStatus) 93, 97, 204

Estado del Trabajo (JobPhase) 96, 97, 203, 204

Estado (Status) 67, 68, 69, 71, 73, 74, 82, 83, 86, 94, 96, 97, 107, 108, 109, 112, 113, 134, 137, 139, 140, 141, 149, 151, 152, 154, 155, 156, 157, 158, 162, 170, 172, 176, 183, 184, 185, 187, 188, 190, 200, 201, 204

Estampación (Stamping) 181, 183, 188

Estandarización 205, 206

Estimación, Presupuesto
 (quotation) 121

Etiqueta de inicio XML 56

Etiqueta final XML 56

Etiqueta (Tag) 40, 56

EXIF 39, 40, 41, 44, 211

Exposición de Planchas
 (PlateMaking) 137

Extensibilidad 20, 22, 23, 24, 41, 45

F

Factura (Invoice) 121

FEFCO 182

Filmación (ImageSetting) 30, 65, 161

FixJDF 198

Flujo de Trabajo 1, 6, 9, 15, 18, 25, 28, 77, 193, 197, 205

Flujo de trabajo híbrido 160

Flujo de trabajo JDF 5, 6, 7, 1, 5, 6, 9, 15, 20, 63, 77, 83, 89, 102, 103, 139, 164, 193, 195, 196, 197, 205

Flujos de trabajo JDF/JMF 3, 5

Folio alto/bajo 207

Fondo de Auditoría (AuditPool) 72, 81, 82, 83, 92, 118, 145

Fondo de cantidad
 (AmountPool) 158, 159

Fondo de Recurso de Enlace
 (ResourceLinkPool) 66

Fondo de Recurso (ResourcePool) 58, 66, 67, 68, 69, 70, 71, 77, 80, 86, 112, 122, 139, 187, 201, 203

Fondo Enlace de Recurso
 (ResourceLinkPool) 113, 139

Fondos de Auditoría (Auditpool) 72

Forma de impresión 147, 150, 153, 208

Forma de Troquel (DieLayout) 183

Formas (Tools) 181

Formato de Mensaje de Trabajo (JMF) 7

Formato de Orden de Trabajo Portátil 39, 50

Formato de Orden de Trabajo Portátil PJTF 39

Formato de Producción de Impresión (PPF) 7, 1, 3, 4, 6, 9, 12, 13, 22, 23, 39, 78, 79, 80, 84, 85, 89, 91, 92, 93, 94, 96, 97, 98, 99, 104, 116, 117, 118, 123, 124, 129, 145, 167, 168, 169, 173, 179, 187, 193, 194, 198, 200, 202, 203, 204, 211, 222

Franja de encolado (GlueLine) 189

Frecuencia de trama 3

Funcionamiento 97

Funciones Dispositivo (Device-Capabilities) 149

G

Generación de previo
 (PreviewGeneration) 75, 112, 139

Gestión de color (ColorManagement) 160, 207

Gestión de flujo de trabajo 14, 16, 17, 18, 24, 50

Grabado de Planchas
 (PlateMaking) 127

Grapado (Stitching) 174

Grosor del papel (Thickness) 155

Groupware 13

Grupo de procesos 174

Guillotina de tres cuchillas (Three Knife Cutter) 169, 171, 174

Guillotina (Trimmer) 1, 2, 9, 49, 131, 167, 169, 171, 174

H

Hermanos (siblings) 56

Hoja o pliego (Sheet) 52, 158

Http 8, 58, 60, 62, 67, 124, 146, 198, 200, 204, 205, 206

Huecograbado (gravure) 145, 148, 153, 159, 160

I

ICS MIS a Acabado 84

ID de referencia (refID) 93, 94

ID de separación nuevo
 (NewSpawnID) 82

ID de separación (SpawnID) 82

Identificación (ID) 86

Identificador de nodo (jRefID) 82

Identificador Universal de Recursos URI 57, 61, 212

Ignorar perfil incurstado
 (IgnoreEmbeddedIC)C 143

Ignorar Imposición PDL
 (IgnorePDLImposition) 165

Imagen Final (FinalImaging) 107

Imponible (Chargeable) 96

Imposición 10, 12, 28, 29, 30, 31, 48, 49, 50, 51, 52, 65, 75, 77, 106, 107, 108, 109, 127, 130, 131, 132, 140, 144, 153, 160,

- 162, 164, 169, 171, 177, 182, 183, 184, 185, 196, 208
- Imposición (Imposition) 160
- Impresión Convencional (ConventionalPrinting) 107, 141
- Impresión de embalajes 7, 124
- impresión digital 10, 33, 74, 140, 160, 162, 163, 164
- Impresión Digital de Oficina 160
- Impresión Digital Integrada (IDF) 160
- Impresión flexográfica 148, 159, 179
- Impresión Offset 34, 146, 148, 150
- Indicador de plegado (FoldIndex) 189
- Información de dispositivo (DeviceInfo) 93, 96, 97, 203, 204, 216
- Información del cliente (CustomerInfo) 122
- Información del dispositivo (DeviceInfo) 93, 96, 97, 203, 204
- Información de nodo (NodeInfo) 80, 81, 98, 117, 118, 185
- Instrucción 15
- Intercambio Comercial (Business2Business) 57
- Interfaz 1, 3, 6, 10, 12, 51, 78, 83, 84, 85, 89, 102, 104, 106, 119, 120, 123, 124, 125, 126, 136, 146, 168, 177, 194, 198, 199, 200, 201, 203, 209
- Interpretación (interpreting) 32, 50, 138, 160
- Interrumpido (Aborted) 68
- Intervalo de Número (NumberSpan) 71
- IPTC 39, 40, 206, 211
- IXIF Schema for IXIF-specific Properties 43
- J**
- JDF Editor 87, 198
- JDF/JMF 1, 3, 4, 5, 6, 9, 12, 13, 22, 23, 39, 78, 79, 80, 84, 89, 98, 104, 123, 124, 145, 167, 168, 169, 179, 187, 193, 194, 200, 202, 203
- JMF ICS 84
- L**
- Leer (read-only) 81
- Leer y escribir (read-write) 81
- Lenguaje de marcado 57
- Lenguaje de Marcado Extensible (XML) 7, 61, 210
- Límite de Color Negro (BlackColorLimit) 52
- Límite de sobreimpresión (StepsLimit) 137
- Limpieza 97
- Lista de Actividades 25
- Lista de Ejecución (RunList) 75, 138, 140, 141, 165, 177
- Lista de Producción (RunList) 77
- Lista de Visualización 33
- Lista Mostrada (DisplayList) 208
- Localizador Universal de Recursos URL 58, 82, 83, 86, 95, 142, 145, 165, 166, 182, 183, 184, 185, 212
- M**
- Mapa de bits 33, 138, 209
- Maquetación (Layout) 133
- Maqueta de imposición 127
- Maqueta (Layout) 75, 77, 154
- Máquina (Machine) 79
- Marca (Brand) 155
- Marcador de posición (placeholder) 133
- Marcas de control (controlmarks) 131, 133, 152, 208
- Marcas (Marks) 139
- Marco Descripción de Recursos (RDF) 59, 206
- Margen de pinza (gripper edge/margin) 29
- Marks 139
- Material Expuesto (ExposedMedia) 76
- Matriz 106, 144, 171, 181, 183, 184, 185, 207, 209
- Matriz Actual de Transformación (CTM) 52, 133, 134, 144, 154
- Matriz (Grid) 184
- Mensaje de cliente (CustomerMessage) 115
- Mensaje JMF 93, 202, 203, 204
- Mensajes de recursos 95
- Metadatos 15, 16, 18, 19, 39, 40, 41, 42, 44, 50, 59, 61, 63, 90, 147, 164, 191
- Método de Pago (CostType) 96
- MIME (Multipurpose Internet Mail Ext) 85, 198, 211
- MIS a Acabado ICS 116
- MIS a Impresión Convencional/Pliego ICS 116
- MIS a Preimpresión ICS 116
- MIS ICS 84, 116, 118
- Modelo proceso/recurso 36, 63, 182
- Modelo productor/consumidor 26, 27, 66
- Molde/Matriz de Separación (Stripping Tool) 181
- Montaje de Caja (BoxFolding) 181
- Montaje (Stripping) 107
- Motores de flujo de trabajo 15, 24, 80
- N**
- NewSpawnID 82, 83
- Nivel 85, 86, 89, 98, 117, 129
- Nodo 45, 47, 63, 64, 65, 66, 68, 69, 70, 72, 75, 76, 81, 82, 98, 106, 107, 110, 113, 115, 118, 198
- Nodo producto 64, 65, 106
- Nodos de proceso 64, 65, 66, 76, 118, 129
- Nombre de Bloque (BlockName) 186
- Nombre de Cliente (CustomerInfo) 145
- Nombre de Hoja (SheetName) 74, 97, 108, 109, 113, 134, 152, 154, 155, 156, 157, 158, 170, 186, 187, 188
- Nombre de Signatura (SignatureName) 74, 97, 108, 109, 113, 133, 134, 152, 154, 155, 156, 157, 158, 162, 187, 188

- N**
- Nombre trabajo de cliente (CustomerJobName) 68, 115, 201, 203
 - Normalización 3, 20, 23, 24, 50
 - NormasDB (DBRules) 175
 - NormasDB o DBRules 165
 - Normas de Tipografías (FontPolicy) 136
 - NumberUp 109, 185
 - Número de cliente (CustomerID) 68, 115, 201
 - Número de orden del Cliente(CustomerOrderID) 115, 201
 - Número de Orden (JobID) 145
 - Número proyecto de Cliente (CustomerProjectID) 115
 - Números de página XMP (Npages) 44
- O**
- Objeto de Marcas (MarkObject) 135
 - Objeto de Negocio (BusinessObject) 120
 - Offset digital 36
 - Opciones de Conversión (ColorSpaceConversionOp) 143
 - Operador del dispositivo (employee) 96
 - Operation 48, 142
 - Orden de compra (PurchaseOrder) 121
 - Order Management System 212
 - Orientación de página (frontpages) 133, 185
 - Original 97
- P**
- Packaging 179
 - Parado 97
 - Parámetros Conversión de Color (ColorSpaceConversionParams) 142
 - Parámetros de Aprobación (ApprovalParams) 141
 - Parámetros de Celda (StripCellParams) 108, 109, 184
 - Parámetros de Combinación (DBMergeParams) 165
 - Parámetros de Encuadernación (BinderyParams) 184
 - Parámetros de Interpretación (InterpretingParams) 32
 - Parámetros de Montaje (StrippingParams) 131, 183
 - Parámetros de Plegado (BoxFoldingParams) 188
 - Parámetros de Procesado (RenderingParams) 33, 77
 - Parámetros de Producción (FormaDef) 182
 - Parámetros de recorte (TrimmingParams) 171
 - Parámetros de Tramado (ScreeningParams) 32
 - Parámetros Impresión Convencional (ConventionalPrintingParams) 112, 148
 - Parte de Trabajo Electrónico (JobTicket) 15, 20, 52, 103
 - Paso y repetición (Step&Repeat) 182, 184
 - Patrón de Plegado (Folding Pattern) 184
 - PDF/VT (Variable Transactional) 164
 - Pedido 3, 9, 11, 12, 16, 17, 62, 65, 78, 96, 103, 119, 121, 132, 147, 156, 174
 - Perfiles de color 143
 - Personalización 163
 - Persona (Person) 56, 57
 - Peso del papel (Weight) 71, 155, 156
 - PipePull 95
 - PipePush 95
 - PJTF 23, 24, 39, 50, 51, 52, 53, 63, 90, 124, 212
 - Plancha (Media) 133
 - Plantilla Combinada (DBTemplateMerge) 165
 - Plantilla de Imposición (SignatureCell) 109, 185
 - Plantilla de MaquetaDB (DBDocTemplateLayout) 165
 - PlateTechnology 179
 - Plegado de Caja (BoxFolding) 188
 - Plegado de Cajas (BoxFoldActions) 188
 - Plegado (Fold) 185, 189
 - Plegado (Folded) 189
 - Plegado (Folding) 107
 - Pliegos (Sheet) 74
 - Postimpresión 1, 37, 167, 168
 - PPF 1, 2, 3, 6, 9, 11, 23, 24, 39, 45, 46, 47, 48, 49, 51, 52, 63, 138, 150, 167, 169, 212
 - PPML 164
 - PPML/GA 164
 - PPML/VDX 164
 - Preajuste de máquina 17
 - Preferencias de Entrega (DroptemIntent) 69
 - Preferencias de Envío (DeliveryIntent) 69, 121
 - Preferencias de Maqueta (LayoutIntent) 70, 111
 - Preferencias de Soporte (MedialIntent) 70, 71, 111, 179
 - Prefijo (prefix) 56
 - Preimpresión 1, 17, 28, 107, 111, 116, 123, 124, 126, 127, 129, 139, 156, 177, 180, 191, 205, 210
 - Preimpresión/Preparación (PrepressPreparation) 107
 - Presna: MIS a Impresión en Bobina ICS 116
 - Preparación de Imposición (ImpositionPreparation) 126
 - Preparación de los datos (PrepressPreparation) 127
 - Preparación de Preimpresión (PrepressPreparation) 126
 - Preparación imposición (ImpositionPreparation) 127
 - Preparado (Ready) 68
 - Previsualización (Preview) 46, 112
 - Print Engine 50, 205
 - PrintEngine2 164
 - PrintTalk 61, 62, 102, 118, 119, 121, 206

- Procesador de ordenes de trabajo (JTP) 50
 - Procesador de parte de trabajo electrónico 15
 - Procesado (Rendering) 32, 50, 103, 138
 - Proceso combinado 64, 65, 68, 75, 77, 139, 141, 146, 161, 174, 177, 181, 191
 - Producción de Troqueles (DieMaking) 181
 - Producción Forma del Troquel (DieLayoutProductionParams) 183
 - Profundidad de color (Color Depth) 33, 48, 208
 - Propósito de Conversión (RenderingIntent) 143
 - Protocolos de mensajería 90
 - Proyecto (Draft) 141
 - Prueba análogica 140
 - Prueba aprobada (ProofApprovalResponse) 121
 - Prueba de color 140
 - Prueba de contacto 140
 - Prueba de contenido 10, 28, 31, 32, 123, 127, 157, 208
 - Prueba de contrato 140
 - Prueba de Imposición (ImpositionProofing) 126
 - Prueba de Página (PageProofing) 127
 - Prueba de pantalla 140
 - Prueba electrónica (Soft Proof) 140
 - Pruebas de Pantalla de la Imposición (ImpositionSoftProofing) 127
 - Pruebas de Pantalla de Página (PageSoftProofing) 127
- R**
- Rango del par XY(XYPairSpan) 71
 - Rasterizado de Imposición (ImpositionRIPing) 137
 - Rasterizado de la Imposición (ImpositionRIPing) 126
 - Rechazado (Rejected) 141
 - Reconocimiento JMF 92
 - Recopilación 174
 - Recorte (Trimming) 160, 171, 174
 - Rectángulos (BoundingBox) 182
 - Recubrimiento del anverso (FrontCoatings) 155
 - Recubrimiento del reverso (BackCoatings) 155
 - Recurso físico 33, 69
 - Recursos abstractos 115
 - Recurso solo lectura (rRefsROCopied) 82, 83
 - Red(es) 1, 6, 9, 12, 13, 19, 62, 66, 89, 90, 120, 157, 160, 187, 194, 195, 196, 197
 - Referencia a elemento (RefElement) 115
 - Referencia a recurso (rRef) 69, 70, 77, 108, 113, 139, 155, 157, 158, 162, 184, 187
 - Registro de fecha (TimeStamp) 72, 82, 83, 91, 93, 94, 204
 - Registro JMF 92
 - Reglas ICS 86
 - Relieve (Embossing) 185
 - Reposo (Idle) 97
 - Respuesta del pedido (OrderStatusRequest) 121
 - Respuesta JMF 92
 - Reventado Imagen/Imagen Imagen/ Objeto (ImageToImageTrapping e ImageToObjectTrapping) 137
 - Reventado (Trapping) 28, 52, 137, 156, 177, 209
 - Reverso (Back) 74
 - Ruta (FileSpec) 86
- S**
- Sección de Ensamblado (AssemblySection) 175
 - Secuencia de color (DeviceColorantOrder) 157
 - Secuencia de contenido (String Content) 203
 - Secuencia (String) 115
 - SelecciónDB (DBSelection) 166, 175
 - Señal JMF 92, 96
 - Separación 81, 181
 - Separar y Combinar (Spawn&Merge) 81
 - Serigrafía 145, 148, 153, 159
 - Serigrafía (silk screen printing) 145
 - Servicios Web 89, 206
 - Signatura de Encuadernación (BinderySignature) 109, 184, 185
 - Signatura (Signature) 52, 74, 109, 153, 184
 - Sistema de flujo de trabajo 11, 12, 19, 20
 - Sistema de Gestión de Flujo de Trabajo 15
 - Sistema de gestión de pedidos (OMS) 12
 - Sistemas de Planificación de Recursos Empresariales (ERP) 101, 211
 - Sobranate (Shingling) 209
 - Sobreimpresión Imagen/Objeto (ImageToObject-Trapping) 52
 - Sobreimprimir 137
 - Software de creación XMP (CreatorTool) 44
 - Solicitud de aprobación (ProofApprovalRequest) 121
 - Solicitud de recurso (ResourceQuery) 95
 - Soporte Expuesto (ExposedMedia) 112, 141
 - Soporte (Media) 111, 113
 - SpawnID 82
 - StatusDetails 97
 - String 115, 203, 204
 - Suscripción (Subscription) 98, 145
 - Sustrato (Media) 76
- T**
- Tabla de planificación 14
 - Tabla referencias cruzadas 51
 - Tamaño (Dimension) 155
 - Terminado (Completed) 68
 - Tiff 1bit 138
 - TIFF-B 31, 138, 139
 - Tinta (Ink) 113
 - Tipo de codificación (encoding) 55

T

Tipo de datos (DataType) 71
Tipo de impresión (WorkStyle) 108, 149, 184
Tipo de Objeto (SourceObject) 143
Tipo de sustrato (MediaType) 154
Tipo de trabajo (WorkType) 97
Tipo Signatura Encuadernación
(BinderySignatureType) 184, 185
Tirada (Pressrun) 6, 13, 14, 103, 112, 201
Trabajador (Worker) 84
Trama AM/FM 33, 209
Tramado (Screening) 32, 50, 138, 139, 160
Trapping 52, 135, 136, 137, 142, 156, 177, 209
Trapping por Defecto (DefaultTrapping) 136
Trazado de corte (CutPath) 182
Trazado (Mask) 184
Trazados de Troquel (DieLine) 184
Troquelado (DieMaking) 178
Troquelar (ShappeCutting) 181
Troquel de corte (CutDie) 181
Troquel (Die) 184

U

UCS 56, 212
Umbral de conversión Gris RGB a Negro. 143
Umbral de conversión Gris RGB/Negro
(RGBGray2BlackThreshold) 143
UTF 55, 56, 58, 67, 162, 204

V

Valores 11, 24, 28, 34, 46, 48, 55, 60, 65, 66, 68,
75, 103, 105, 109, 113, 122, 125, 131, 132, 135,
138, 143, 144, 147, 149, 150, 151, 156, 167,
171, 173, 179, 182, 186, 188, 193, 195, 207
Valores preajuste zonas de entintado (InkZoneProfile) 151
Versión ICS 118
Versión (Version) 60, 67, 68, 94, 162,
187, 201, 204, 205, 206, 210
Vías estables de comunicación (PersistentChannel) 97
Visual Basic 126, 200, 201
Voltrear (WorkAndTurn) 149

W

Web to Print 61, 119
Workflow Management Coalition (WFMC) 14

X

XML 7, 12, 23, 42, 43, 51, 55, 57, 59,
60, 61, 63, 67, 68, 71, 87, 90, 91, 92,
164, 165, 203, 205, 206, 210
XML Paper Specification 57, 213
XMP 15, 39, 40, 42, 43, 44, 52, 54, 59, 60, 61, 205
XSD 58, 213

Z

Zonas de entintado 21, 28, 31, 32, 45, 46,
47, 50, 112, 150, 152, 159, 177

Autores

Professor Dr. Thomas Hoffmann-Walbeck: Matemático, desarrollador de software y director de proyectos. Profesor de tecnologías de impresión y medios en la Universidad de Stugart desde 1998; áreas de preimpresión, computación aplicada y redes JDF.

Sebastian Riegel: Licenciado en Ingeniería de impresión y medios Universidad de Stugart. Asistente científico desde 2003 en dichas materias. Responsable técnico del laboratorio de investigación CTP y director del programa de la integración JDF de la universidad.

Traductores

John Joseph (Joe) Dobrowits: Responsable de la traducción alemán inglés. Licenciado en lengua alemana y literatura, máster certificado de postgrado en sistemas gestores de información y proyectos y comercio electrónico. Durante 20 años coordinación tecnológica impresión/marketing. Gestor de impresión Milwaukee Wisconsin.

Miguel Eduardo Menéndez Romero: Responsable de la traducción inglés español/castellano. Estudios Universidad Laboral de Cheste, inglés/computación San Diego California. Formador en diseño, maquetación, ilustración, pre/impresión y colorimetría. Plan formativo de artes gráficas Apeg/Coepa 2004. Miembro de CIP4 y PIA/GATF desde 2006. Asesor gráfico.

PIA Industrias Gráficas de América

Printing Industries of America PIA

Graphic Arts Technical Foundation GATF

La fusión en 1999 de PIA (1887) con GATF (1924) supuso la creación de la mayor organización mundial de comercio para Industrias Gráficas representa y sirve a los intereses de 10.000 empresas asociadas y más de un millón de empleados. Desarrolla productos y servicios para el impulso, rentabilidad y eficacia de afiliados e industrias de la comunicación. Su personal divulga educación, investigación e información a más de 80 países en tareas de control de calidad tecnológica/ambiental e incremento de la productividad a través de laboratorios, conferencias, publicaciones y servicios.

Sus miembros son empresas relacionadas con el mundo de la impresión e incluyen a usuarios y proveedores de numerosos ámbitos como la impresión convencional, digital, rotativa, embalajes, postimpresión, desarrollo de aplicaciones y servicios gráficos, ventas y marketing. .

Printing Industries Press publica libros, audiovisuales, informes financieros y tecnológicos y la revista bimensual *Printing Industries of America The Magazine*, insignia de la asociación que informa sobre temas legislativos, científicos, industriales, económicos y humanos.

Más información en www.printing.org

Asociados de Printing Industries of America

Canadian Printing Industries Association

Ottawa, Ontario
www.cpia-aci.ca

Graphic Arts Association

Trevese, PA
www.gaa1900.com

Pacific Printing and Imaging Association

Portland, OR
www.ppiassociation.org

Printing & Graphics Association

MidAtlantic
Columbia, MD
www.pgama.com

Printing & Imaging/Association of MidAmerica

Dallas, TX
www.piamidam.org

Printing & Imaging Association of Georgia

Smyrna, GA
www.piag.org

Printing Association of Florida

Orlando, FL
www.flprint.org

Printing Industries Alliance

Amherst, NY
www.pialliance.org

Printing Industries of Arizona/New Mexico

Phoenix, AZ
www.piaz.org

Printing Industries Association of San Diego

San Diego, CA
www.piasd.org

Printing Industries Association Inc. of Southern California

Los Angeles, CA
www.piasc.org

Printing Industries of Ohio • N. Kentucky

Westerville, OH
www.pianko.org

Printing Industries of America Mountain States

Greenwood Village, CO
www.printincolorado.org

Printing Industries of the Gulf Coast

Houston, TX
www.pigc.com

Printing Industries of Michigan, Inc.

Southfield, MI
www.print.org

PINE

Southborough, MA
www.pine.org

Visual Media Alliance

San Francisco, CA
www.visualmediaalliance.org

Printing Industries of St. Louis, Inc.

Maryland Heights, MO
www.pistl.org

Printing Industries of Utah

West Jordan, UT
www.piofutah.com

Great Lakes Graphics Association

Pewaukee, WI
www.glga.info

Printing Industry Midwest

Roseville, MN
www.pimn.org

The Printing Industry of the Carolinas, Inc.

Charlotte, NC
www.picanet.org

Printing Industry Association of the South

Nashville, TN
www.pias.org

La Organización CIP4

CIP4 Organización de Cooperación Internacional para la Integración de Pre/Post Impresión sin ánimo de lucro que normaliza la automatización en la industria gráfica con representación en 31 países de todo el mundo; sus miembros son empresas de preimpresión, editores, imprentas, consultores, distribuidores de software, proveedores de sistemas gráficos, educadores e integradores.

Los 300 miembros registrados aproximados reúnen unas 1.600 personas integradas en 17 grupos que trabajan en temas técnicos y educativos para promocionar la automatización. Cualquier empresa dedicada a las artes gráficas puede ser miembro de CIP4. Cada año se celebran dos congresos *Interops* presenciales para tratar la *Interoperabilidad* donde se redactan los tutoriales técnicos para miembros y grupos de trabajo. Las demás "reuniones" se convocan a través de Internet por comodidad. Las reuniones presenciales son muy importantes para la divulgación y una oportunidad para comprobar los sistemas y comentar los resultados en directo.

CIP3, JMF, JDF y PrintTalk son beneficios que CIP4 aporta a la industria pero sus miembros disfrutan de otros recursos:

- Kits de desarrollo para programas.
- Sistemas para la comprobación del software.
- Utilidades como *CheckJDF* y *JDF Editor*.
- Grupos de Usuarios Internacionales.
- Programas de certificación del producto.
- Programa de formación *Experto Certificado JDF*.
- Grupo Redes de Negocios - Asociación para el desarrollo de programas y mercados.
- El mercado JDF - Directorio de productos y servicios que soportan JDF y PrintTalk.
- La Matriz de Integración JDF - Referencia de integración de dispositivos en la imprenta.
- Estudios de casos instalados.
- El Boletín JDF - disponible seis veces al año para todo el público.

En www.cip4.org puede encontrar todo esto y mucho más.

Títulos relacionados



Guía Práctica para Impresión Automatizada

James E. Harvey ASN (Traducción Miguel Eduardo Menéndez Romero)

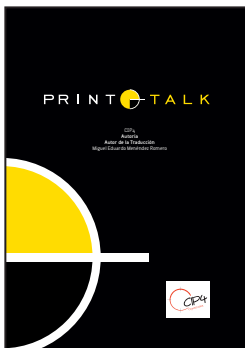
Traducción al castellano de la Guía de Soluciones de Automatización, anteriormente Grupo de Usuarios JDF. Desde el año 2008, ha sido un punto de encuentro para impresores y proveedores con el objetivo común de la automatización integral de la impresión. Los miembros de la ASN son imprentas grandes y pequeñas, familiares o nacionales, de offset de hoja/bobina y digitales. Toda la experiencia recopilada en esta publicación le ayudará a tomar las decisiones correctas.

Es una reflexión sobre los beneficios, desafíos y métodos para integrar la automatización en su empresa o negocio, ofreciendo datos sobre la relación con proveedores, clientes, operarios y directivos. Las consideraciones operativas y sugerencias de esta publicación resultarán de gran utilidad si piensa seriamente en la automatización de su imprenta.

www.printing.org

www.todoebook.com

www.miguelsenendez.com



PRINTALK

CIP4 (Traducción Miguel Eduardo Menéndez Romero)

Estándar XML de la Industria Gráfica para el intercambio de información comercial

PrintTalk es un estándar XML que proporciona un formato único a los Proveedores de Impresión para comunicar Transacciones Comerciales y especificaciones de Productos de Impresión tanto a sus Clientes como entre ellos mismos.

El objetivo principal de esta guía es describir el protocolo PrintTalk a los programadores que deseen incluir en sus productos de software la especificación de interfaz PrintTalk, aunque a los impresores les serán útiles algunas partes de esta especificación para profundizar en el conocimiento del comercio electrónico de sus productos gráficos. Para comprender plenamente este documento es aconsejable conocer XML y estar familiarizado con JDF.

El libro incluye ejemplos de transacciones y comunicación mediante código XML

www.printing.org

www.todoebook.com

www.miguelsenendez.com

FLUJO DE
TRABAJO

JDF

GUÍA DE AUTOMATIZACIÓN PARA INDUSTRIAS GRÁFICAS

JDF y JMF están basados en el estándar XML y son la innovación más importante para la automatización de la industria gráfica. Esta guía demuestra que son más que simples formatos de transferencia de datos proporcionando un exhaustivo análisis de sus componentes y de los flujos de trabajo que se pueden implementar con su ayuda.

- Contenidos
- Características de JDF
- Escenarios de producción
- Modelos proceso/recurso (productor/consumidor)
- Introducción a XML
- Estructuras JDF
- Formato mensajes de trabajo JMF
- Flujos de trabajo y equivalentes JDF

Ejemplos de instalaciones JDF incluyendo sistemas gestores de pedidos MIS, preimpresión, impresión, acabados y embalajes. Es un excelente manual de consulta para aprendices, estudiantes, desarrolladores y usuarios de tecnologías de impresión.

Thomas Hoffmann-Walbeck es matemático, director de proyectos, programador y profesor en la Universidad de Stúgart

Sebastian Riegel ingeniería de impresión y medios, responsable del laboratorio CTP y director de integración JDF en la Universidad de Stúgart

Miguel Eduardo Menéndez Romero asesor gráfico portada, traducción y edición 2013*

