LaTeX technologies at work — aesthetically beautiful PDFs on the fly from XML input: XML Page Composition (XPC) micro-service in the cloud

Rishikesan Nair T., Aravind Rajendran, Rajagopal C.V., Radhakrishnan C.V.

Abstract

XML Page Composition (XPC) is a micro-service in the cloud which is built on the web-based typesetting framework TEXFolio from River Valley Technologies, India, a typesetting technology company established in India in 1996 by the brothers C.V. Radhakrishnan, C.V. Rajendran and C.V. Rajagopal, which acts as a technology provider for STM Document Engineering Pvt Ltd (STMDocs), which in turn uses TEX and friends for typesetting and provides prepress services to leading publishers around the world.

The purpose of XPC is to automate PDF creation from the XML source using an automated workflow without any manual intervention. Of the several recently developed web-based frameworks by River Valley Technologies India, XPC is the newest. Ithal [1], Neptune [2] and TEXFolio [3] are other milestone developments of River Valley Technologies. A few more products and services specifically focussing on empowering the author are under development at River Valley.

A valid XML along with the graphics and other metadata files associated with it should be made available to the XPC system to generate a PDF. An automated quality control (QC) process is performed on the PDF output and a number of parameters, both standard typesetting specifications and publisher-specific requirements, are checked by the system itself as part of the final validation.

1 Introduction

Prepress work for scientific, technical, and medical (STM) journal production has been subjected to enormous changes over recent years, in order to meet growing technology requirements, speed up the production process, and reduce overall production time. The aim is to publish articles as quickly as possible, thus reducing manual labour to increase accuracy and cost reduction. In the beginning, the research articles or other materials were typeset for print media only. However, when the Internet came into the picture the landscape radically changed. The requirement for many different types of outputs become a de facto standard, and the typesetter who does the prepress work has to generate SGML/XML/MathML and web-optimized PDFs in addition to the "fat" PDF or print PDF, all from a single source which the author provides.

The current scenario is that a publisher uses typesetting services either from one prepress supplier, or a few, distributing its journals among them. Those supplier(s) are responsible for the entire production of the particular journal(s) assigned to them:

(1) media conversion,
(2) file structuring,
(3) copyediting,
(4) producing proofs for authors,
(5) incorporating author corrections,
(6) producing XML/MathML, web optimized PDFs, print ready PDFs and electronically publishing them for article-based publishing,
(7) compiling the articles into a journal issue, per the instructions from the publisher.

Leading publishers of STM journals are recently thinking along new lines, trying to distribute the prepress work of even a single journal to many typesetters. For example, steps (1) to (5) to the first supplier; (6) to a second supplier and (7) to a third.

XML Page Composition (XPC), a new product from River Valley Technologies India[1] which STMDocs[2] has evaluated can play a major role in the prepress work industry. XPC is deployed under stage 6 (see above) in a fully automatic mode. Now let us look at XPC in detail.

2 XML Page Composition Service (XPC)

The XPC micro-service is a typesetting system in the cloud to create standards-compliant and aesthetically pleasing PDF using TEXFolio, directly from a valid production XML, assets and metadata. (TEXFolio is the TEX-based typesetting framework in the cloud.)

The Automated Quality Control system (Auto-QC) built into XPC ensures the quality of the generated PDF output and also carries out publisher-specific validation. Auto-QC is based on certain rules and standards which are predefined. Column balancing, float placement, overfull boxes, and underfull boxes are a few of the issues checked by auto-QC. Auto-QC produces an error report in PDF format for the operator.

Currently numerous templates for one of the leading STM publishers are configured. There is no limit on the number of typesetting models that can be configured.

All files will pass through XPC without manual intervention. However, heavy math, depending on

[1] http://www.river-valley.com
[2] https://www.stmdocs.in

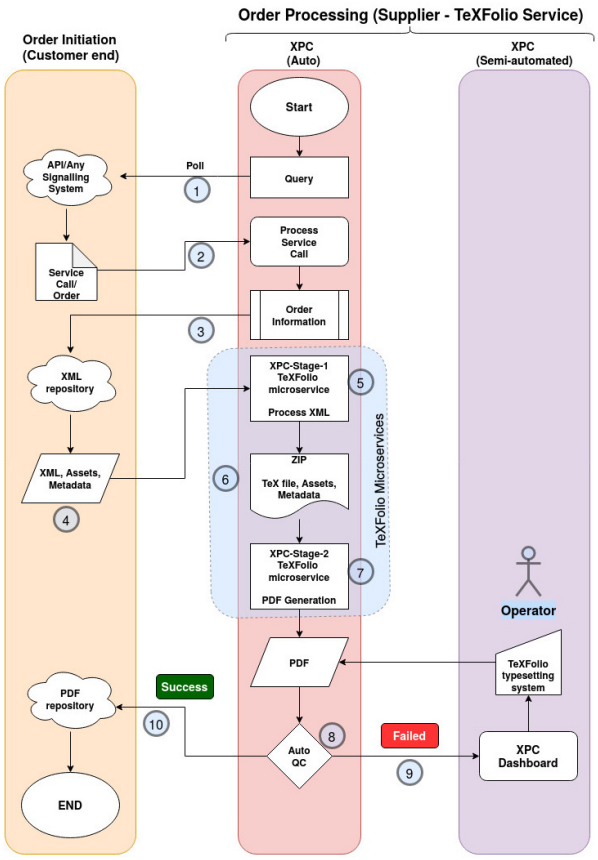

Figure 1: XPC workflow.

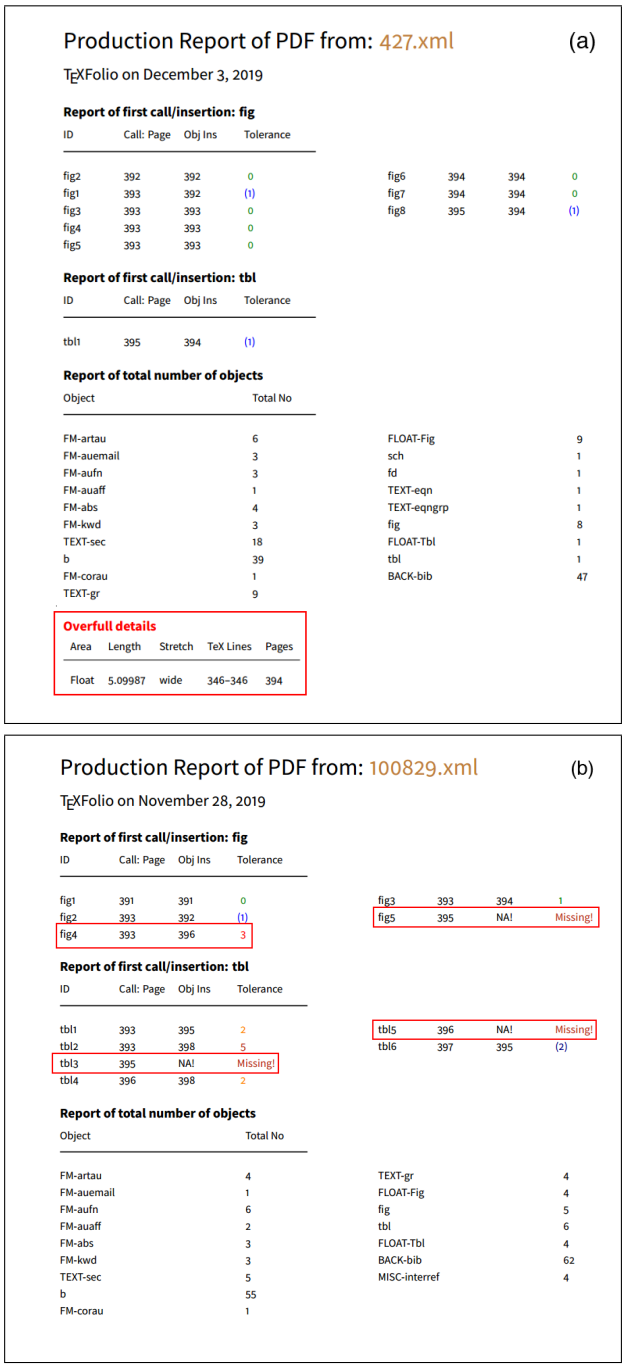

Figure 2: Auto-QC error reports.

the needs of the publisher, may need manual support, mostly with pagination.

2.1 XPC Features

The main features of XPC (the figures are grayscaled for print):

1. XPC workflow (Fig. 1) is in the cloud.
2. No content editing or alterations of the source, hence no accidental human errors.
3. High-level automated QC between PDF output and XML, as described. Two sample reports are shown in Fig. 2.
4. Formatting/pagination of PDF output, if required, is done using a control file generated from the XML, without touching the XML data.
5. Application of artificial intelligence for table formatting and float placement, thereby reducing manual effort.
6. Multilingual support, currently configured for 11 languages.
7. Automatic table width calculation to help typeset tables in either single column or double column mode without any processing instructions.
8. APIs to support command-line operations for automation.

2.2 Workflow

The workflow diagram in Fig. 1 along with the explanation of each step provided in the following will describe in more detail the functionality of the XPC service.

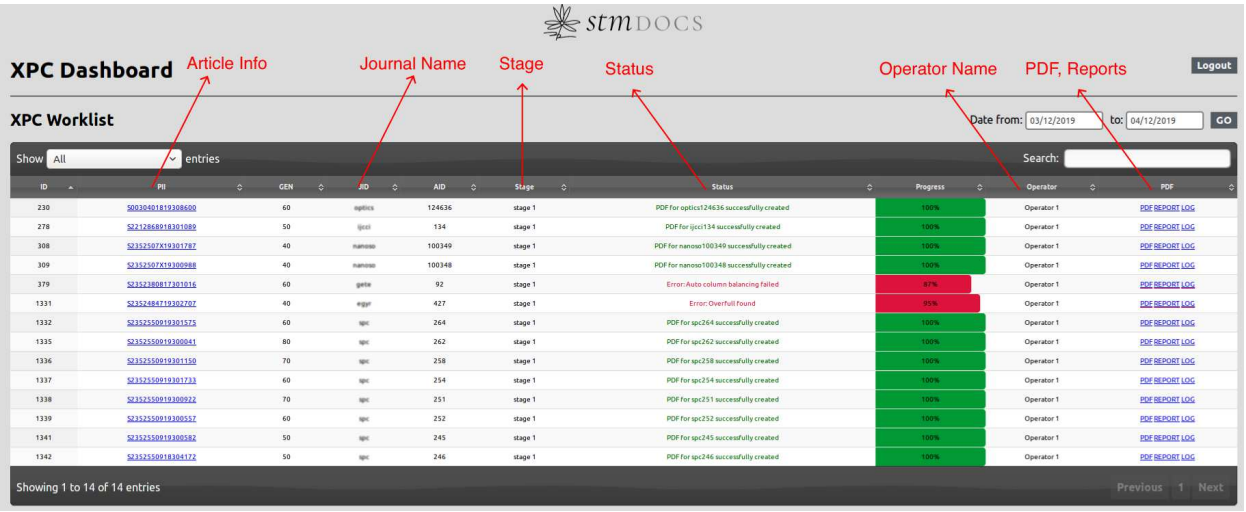

Figure 3: XPC operator dashboard as implemented at STMDocs during evaluation.

1. Order initiation: API service call or check for orders in publishers' servers.
2. Process service call or orders, and extract order information.
3. Send order information to data repository or datastore.
4. Retrieve input XML and assets of the items which are generated by the XML supplier.
5. Process XML: Add namespaces, find table width, create an external float control file and create a TEX file. Find the typesetting model and (only if a journal with a new model is received) create a typesetting template configuration file automatically.
6. Make an archive of all these and push them to TEXFolio microservice for processing.
7. Create PDF.
8. Trigger auto-QC: If the quality of the PDF output is not up to the benchmark or publisher specifications, flag a failure, likely a need for manual pagination.
9. Items requiring manual intervention get listed in the operator's dashboard (see Fig. 3). Operator paginates using the float control file, pushes the finished PDF again for auto-QC.
10. If auto-QC is successful, PDF will be delivered to the client.

2.3 Error reports — Details

Two error reports are given in Fig. 2 (above and below are two separate reports). There are three main sections which will appear in every article which contains figures and tables: "Report of first call/insertion: fig", "Report of first call/insertion: tbl", "Report of total number of objects". The optional section "Overfull details" will appear only if the PDF output has any overfull text.

One of the many challenges of the auto-pagination function is the placement of floats near their references. XPC will do a fairly nice job here, however in very rare cases due to severe constraints such as a small number of pages, a large number of floats, and a two-column document, as one can imagine, it is a difficult task to place the floats near to their references even manually. If the floats are placed far from their citations, this information will be flagged in the report and the operator who checks the report can find and correct it. As you can see in the sample reports, the following details are included to help the operator to find the problem:

- `ID`: The ID of the float to search.
- `Call Page`: The page in the PDF where the float is first cited.
- `Obj Ins`: The PDF page where the float is inserted.
- `Tolerance`: The tolerance with which this can be allowed. As the tolerances become worse, they are highlighted with colour changes, red being the worst.

2.4 Issues and challenges

The developers faced many challenges during the development of XPC. A few of them are listed here:

1. Finding journal typesetting model
2. Table cell width calculation
3. Float placement
4. Handling built-up accents

5. Pagination using a control file automatically generated from XML
6. Automated QC of the PDF output
7. Automated column balancing of the last page in a two-column article
8. Automating manual fall-out: Adding more pagination commands in control file — *In Progress*

All except the last have been resolved.

2.5 Estimated production capacity

The service levels presented as part of the pilot phase of the service were these:

1. 30% of the articles can be delivered within 6 hours.
2. 60% of the articles can be delivered within 12 hours.
3. 100% of the articles can be delivered within 24 hours.
4. Capacity that can be handled would be 250 articles per day (average 15 pages/article).

However the performance has been greatly enhanced after the pilot phase. Currently the service capacity is over 600 articles per day (average 15 pages/article).

3 Acknowledgement

STM Document Engineering (STMDocs) gratefully acknowledges and thanks River Valley Technologies for allowing their products and technologies (XPC, TEXFolio, NEPTUNE, Ithal, etc.) to be showcased at TUG. Any credits are to be attributed to River Valley Technologies. We also highly acknowledge the team at STMDocs who have helped with the evaluation of the XPC system, especially Apu V and Rahul Krishnan S and our testing team Akshay K.S. and Sangeetha V.

References

[1] Ithal. https://ithal.io/main.html

[2] R. Aravind Rajendran, Rishikesan Nair T. NEPTUNE — a proofing framework for LaTeX authors. *TUGboat* 40(2):150–152, 2019. https://tug.org/TUGboat/tb40-2/tb125rajendran-neptune.pdf

[3] R. Rishikesan Nair T., Rajagopal C.V. TEXFolio — a framework to typeset XML documents using TEX. *TUGboat* 40(2):147–149, 2019. https://tug.org/TUGboat/tb40-2/tb125rishi-texfolio.pdf

◇ Rishikesan Nair T.
Aravind Rajendran
STM Document Engineering Pvt. Ltd.
River Valley Campus, Mepukada
Malayinkil
Trivandrum 695571
India
rishi (at) stmdocs.in,
aravind (at) stmdocs.in
https://stmdocs.com

◇ Rajagopal C.V.
Radhakrishnan C.V.
JWRA 34
Jagathy
Trivandrum 695571
India
cvr3 (at) river-valley.org,
cvr (at) river-valley.org
http://river-valley.org