



# Produce & Publish Cloud Edition

*Concept for the next generation version of  
Produce & Publish in the cloud*

This document discusses the architecture and concept for the next generation version of the “Single-Source Multi-Channel Publishing” solution Produce & Publish by ZOPYX Limited.

## Background

Produce & Publish is modular Plone-based “Single-Source Multi-Channel” publishing solution that focuses on delivering a full-fledged publishing solution under the one hood.

The current functionality covers:

- conversion of DOCX documents to X(HTML) and import into the Plone CMS
- web-based management of content
- through-the-web management of conversion resources like templates, images, stylesheets and fonts
- one-click conversion and publishing solution (“easy mode”)
- through-the-web configuration of conversion related parameters
- iterative conversions
- single-click publishing
- generation of high-quality PDF documents
- generation of EPUB and other eBook documents
- generation of other office formats like RTF, ODT, DOC possible

Produce & Publish has been released for the first time in 2008 and evolved over several years into the current Produce & Publish Version 2.1 which is compatible with Plone 4.0, 4.1 and 4.2.



# Outline of the new Produce & Cloud edition

## Internal CMS

Plone will be used as primary content-management system and web interface for Produce & Publish. Compared to Produce & Publish version 1 and 2 the new cloud edition will not use the native content-type functionalities of Plone for storing editorial content (XML, XHTML documents, images). In the next version Plone will be loosely coupled with the Produce & Publish core API that will support arbitrary cloud storages like Dropbox, S3 or Google Drive for storing the editorial content

## Cloud Storage

It is a common requirements that Produce & Publish users want to edit content with a working editor. Most web-based editors like (ckEditor, TinyMCE etc.) have various issues - especially when it comes to editing large files, table handling etc. So we learned that we must provide an integration with desktop-based browsers like native HTML/XML editors (e.g. professional XML editors like Doxygen). This requires that the editorial content is available in some way on the local filesystem of the user. Nowadays cloud storages like Amazon S3, Dropbox oder Google Drive provide a seamless integration with the Windows desktop or MacOSX Finder. Produce & Publish editorial content will appear like locally stored content. Synchronisation is handled by the underlying cloud bindings automatically in the background.

## Intranet storage

Produce & Publish Cloud Edition can also be deployed as a traditional intranet application inside your corporate environment. Company employees and editors can access editorial content through their desktop by mounting the related filesystem share for example through Samba

## Produce & Publish in the Cloud

Draft - 24.04.2013

Author: Andreas Jung, ZOPYX Limited, [www.zopyx.com](http://www.zopyx.com)

[www.produce-and-publish.com](http://www.produce-and-publish.com)



(Windows, MacOSX).

## Client-side editing

Produce & Publish Cloud Edition will support editing of editorial content using arbitrary editors like Doxygen, Arbortext etc. on the desktop (if the Produce & Publish server is configured to use a particular cloud storage).

## Through-the-web editing

Instead of using a typical open-source WYSIWYG editor like ckEditor or TinyMCE we will focus on the integration of the commercial “[edit-on NG](#)” editor by the company RealObjects in Saarbrücken, Germany. “edit-on” is implemented in Java and provides a rich API and rich functionality for implementing an easily customizable and easy-to-use professional editor environment through the web. “edit-on NG” supports XHTML.

## PDF conversion

Historically Produce & Publish used the external PrinceXML PDF converter (XHTML/XML + CSS -> PDF). For the next revision of Produce & Publish we will focus on using the “[PDFreactor](#)” by the company RealObjects in Saarbrücken, Germany. Like PrinceXML, PDFreactor supports the generation of high-quality PDF documents from XHTML or XML. It supports CSS3 for styling the content. It provides some extra functionality over PrinceXML like tagged PDFs and encryption. Another high-end PDF converter option is Antenna House V 6.0 (support HTML/XML + CSS).

## EPUB(3) generation

In the planning. Besides the conversion of EPUB and other eBook formats we consider the integration of other professional EPUB conversion pipelines through a webservice API.

## Versioning

Versioning affects editorial content and published content. The versioning strategy and implementation depends on the existing mechanism of the underlying storages used for the editorial content and published content (e.g. Amazon S3 supports versioning out-of-the-box).

## Image handling

## **Produce & Publish in the Cloud**

Draft - 24.04.2013

Author: Andreas Jung, ZOPYX Limited, [www.zopyx.com](http://www.zopyx.com)

[www.produce-and-publish.com](http://www.produce-and-publish.com)



Images will be stored directly together with the primary content (on the same level). The referring content may hold information about image information like scaling information or imagemaps.

## **DOCX conversion and import**

Importing author manuscripts is a common requirement. Right now Produce & Publish uses OpenOffice (connected through a webservice) to convert DOCX to HTML/XHTML. The conversion works more or less. For a more professional and reliable conversion we consider using an externally hosted DOCX conversion service (already used in various commercial projects). The external conversion service provides conversion to Docbook, EPUB and XHTML.

## **Publishing**

The Produce & Publish cloud edition will support publishing content either to the local content-management-system (Plone) or arbitrary cloud storages like Dropbox, Amazon S3 or Google Drive or support the transport to remote hosts using SCP, SFTP etc.

## **Resource handling**

Conversions to PDF or EPUB require a set of associated resources like templates, assets (images, fonts) and styles to convert the media-neutral content. Historically Produce & Publish used to store those assets in the filesystem and import it into Plone in order to edit and maintain resources through-the-web. This approach did not work very well in reality. Integrators prefer maintaining resources on the filesystem (integration with a source code repository is often requested). End-users typically can not deal with stylesheets (requiring CSS knowledge). So there is basically no need for maintaining resources through-the-web. The concept of filesystem based resources as introduced with Produce & Publish version 2 will be carried forward and used the same way for the Produce & Publish Cloud Edition.

## **Produce & Publish server**

Produce & Publish uses a client-server based approach for implementing the interaction with external converters (PrinceXML, Calibre etc.). The current Produce & Publish server implementation works synchronously (client blocks until the conversion has finished). In a recent project we introduced an asynchronous worker model (using an internal queue and asynchronous workers). This functionality is subject to be re-integrated into a later version of the Produce & Publish server for better scalability and more seamless client-side integration.

## **Produce & Publish in the Cloud**

Draft - 24.04.2013

Author: Andreas Jung, ZOPYX Limited, [www.zopyx.com](http://www.zopyx.com)

[www.produce-and-publish.com](http://www.produce-and-publish.com)



## **General architecture**

As with the recent Produce & Publish versions, the overall architecture will remain very modular and pluggable. In fact the modular approach will be carried forward. Especially the so call “Produce & Publish Authoring Environment” is currently heavily tied to the CMS Plone. Although Plone will remain the primary platform as CMS and web-frontend we will move out a bunch of Plone specific (and Plone-related problems) out of the Produce & Publish core into a thin Python layer which is unrelated with Plone). Reusability and modularity are the design goals for the new cloud edition of Produce & Publish.

## **Security & Privacy**

Produce & Publish can be either installed inside your closed company network as an intranet solution or provided as a cloud service. Security and privacy standards depend on the chosen deployment environment. But in all case Produce & Publish Cloud Editon will use the highest possible standards and technologies in order to provide the best level of security and privacy.

## **Contact**

**ZOPYX Limited**

c/o Andreas Jung

Hundskapfklinge 33

D-72074 Tübingen, Germany

[info@zopyx.com](mailto:info@zopyx.com)

[www.produce-and-publish.com](http://www.produce-and-publish.com)

[www.zopyx.com](http://www.zopyx.com)