W3C Widgets for Cloud Resilience : the UNG Project

Smets Jean-Paul
Nexedi
270 Bd Clémenceau,
59700 Marcq-en-Baroeul, France
Telephone number, incl. country code

jp@nexedi.com

Ivan Tyagov
Tyagov IT
lavor 2 Str,
Kuklen 4101, Bulgaria
Telephone number, incl. country code

ivan@nexedi.com

Sébastien Robin
Nexedi
270 Bd Clémenceau,
59700 Marcq-en-Baroeul, France
Telephone number, incl. country code

seb@nexedi.com

ABSTRACT

The UNG project uses W3C Widgets technology to improve the resilience and privacy of modern Cloud by reviving original ideas of 1989 Web. A prototype implementation of an open source Office suite, of an open source e-commerce site and of an open source ERP, developed as W3C Widget and HTML5 Application with offline support, are demonstrated. JIO, a unified API to access and extend various local or remote content storages through the Web, independently of their native APIs, is proposed. The UNG project is supported by a community of open source SMBs, EADS Astrium, Sagem Morpho, INRIA, Institut Télécom, CEA and University of Paris 13.

Categories and Subject Descriptors

E.2 **[Data Storage Representations]**: Composite structures.
H.3.2 **[Information Storage and Retrieval]**: Information Storage.
H.5.2 **[Information Interfaces and Presentation]**: User Interfaces - Windowing systems
H.5.4 **[Information Interfaces and Presentation]**: User Interfaces – Hypertext/Hypermedia
I.7.1 **[Document and Text Processing]**: Document and Text Editing - Document management.

General Terms

Design, Economics, Reliability, Experimentation, Security, Standardization.

Keywords

Browsers, Plugins and Development Tools
Cloud Platforms
Distributed Data Processing
HTML5 and Beyond
The Web on new Devices
Web Standards and Protocols
Resilience
W3C Widgets

1.   INTRODUCTION

The Web in 1993 – at a time NCSA Mosaic was the most popular browser – used to be a decentralized hypermedia system made of static HTML pages and media (images, sounds, videos, files, etc.). It was deployed on neutral IPv4 networks where every workstation could act both as a client and as a server, as a consumer and as a producer of content, thanks to the absence of network address translation (NAT). From time to time, one Web server was no longer accessible, usually for a short period of time. During that time, personal caches such as V6 [1] could be used to access a local copy of important pages while the server was down. Overall, thanks to its socially decentralized architecture and thanks to the mirroring of important sites, the Web used to be a very resilient system which could survive censorship, political crisis, wars and natural disasters.
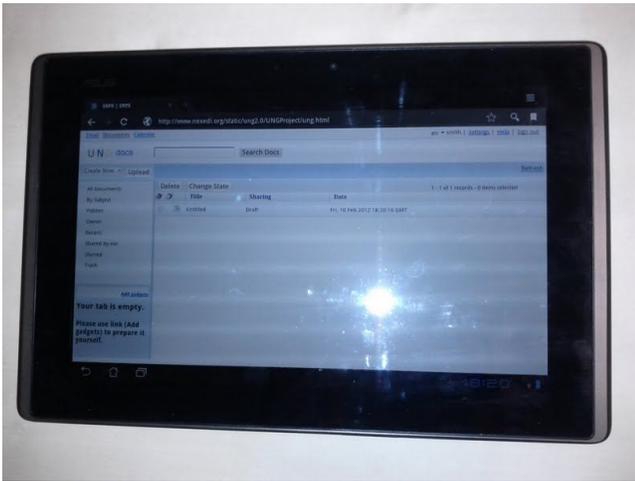
Nowadays, the Web is much more user friendly than in 1993. It can be used to meet the requirements of complex enterprise applications such as ERP [2], CRM [3] and productivity [4]. However, the Web is nowadays much less resilient than it used to be, especially in case of *force majeure* events. Many Web services are now operated on centralized Cloud architectures (socially or technically). The International Working Group on Cloud Resilience [5] has shown that in average, every major provider of Cloud is out of service during one full day per year. In 2011, when Amazon EC2 was down during 4 days [6], tens of thousands of Web sites became unavailable at once.. During the outage, there was no option to maintain a cache or a mirror because of the very dynamic nature of most modern Web sites as opposed to the static nature of Web sites in 1993.

Our presentation shows how W3C Widgets can be used to combine the best of the Web: the resilience of 1993 Web based on static pages with the user friendliness of 2012 Cloud based Web applications. Our presentation will be based on the demonstration of 3 prototypes, each of which exhibits key principles to achieve Cloud resilience through W3C Widgets.

2.   PROTOTYPE 1 : UNG Docs

UNG Docs is an open source Web office suite which includes spreadsheet, illustration and word processing modules. UNG Docs 1.0 was introduced in Brazilia in 2010 at CONSEGI conference on Cloud Computing for e-government. It is a server-side implementation of UNG based on the ERP5 framework [7]. Although UNG 1.0 works fine – and is now integrated to ERP5 – it does not address resilience requirements and can not be operated offline.

UNG Docs 2.0 was introduced in 2011. It is a client-side implementation of UNG Docs. UNG Docs 2.0 can be deployed in different ways: as an HTML5 Application served through HTTP, as a mobile application packaged with PhoneGAP or as a W3C Widget. The same code can be used on about any platform: Android, GNU/Linux, MacOS, Windows, Firefox, Chrome, Opera, etc. Once the code is deployed, UNG Docs can be used offline. If the http server is down, UNG Docs can still be used, which makes it more resilient than for example Google Docs.

UNG Docs 2.0 introduces an API named JIO which stands for Javascript Input/Output. The JIO library abstracts all access to storage so that developers only need to code their application once whatever the target storage they plan to deploy it on. Users can then define how they would like their personal content to be stored through a JSON configuration file named **jio.cfg**. JIO currently supports browser's localStorage and WebDAV backends. JIO also provides so-called virtual storages which extend the features of any backend. JIO's encryptedStorage encrypts/decrypts content before storing it. JIOS's duplicateStorage can save content on two backend storage at the same time.

```
{"type":"index","storage":
 {"type":"encrypt","storage":
   {"type":"asynchronous","storage":
     {"type":"dav","userName":"yourName",
      "password":"yourPassword",
      "location":"storageCompleteURL",
      "provider":"notAvailable"}},
      "password":"yourEncryptPassword"}}
```

UNG Docs 2.0 includes optional support for the unhosted protocol [8]. We found that Unhosted provides a nice way for users to save their jio.cfg preference file and access it from any device without having to remember too many passwords and without having to carry a physical device (USB key, smartcard). UNG Docs could also support Unhosted's remoteStorage as a backend part of JIO. However, the Unhosted protocol is still too much *hosted* in our opinion since it requires online web services. We thus found that the Unhosted protocol is not suitable for mobile or offline usage. It is not suitable either to achieve resilience whenever most users rely on very few webfinger providers. We are thus considering currently how to wrap or extend Unhosted to manage application preferences rather that storage itself.

3.   PROTOTYPE 2 : UNG Commerce

The second prototype we will demonstrate is an e-commerce application based on the UNG Docs framework. In this prototype, we use ERP5 to generate a static web site which consists of a catalog of products. The site hierarchy is as follows:

http://www.ung-project.org/ecommerce/
http://www.ung-project.org/ecommerce/pc/
http://www.ung-project.org/ecommerce/pc/laptop/
http://www.ung-project.org/ecommerce/pc/laptop/thinkpadT400
http://www.ung-project.org/ecommerce/pc/desktop/
http://www.ung-project.org/ecommerce/pc/desktop/
http://www.ung-project.org/ecommerce/phone/

http://www.ung-project.org/ecommerce/phone/android/
http://www.ung-project.org/ecommerce/phone/android/nexus
http://www.ung-project.org/ecommerce/phone/ios/
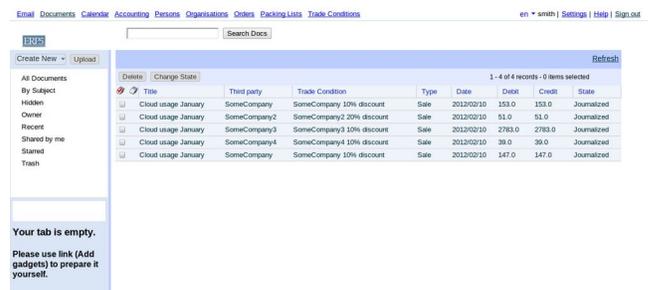http://www.ung-project.org/ecommerce/phone/ios/iphone4s

Every page in the site uses HTML5 standard, rather than JSON or XML, so that it can be parsed by Web search engines or by Web browsers. The (simplified) code of a product page looks like this:

```
<html>
  <head>
    <script src="require.js" type="text/javascript"></script>
    <script src="ung-engine.js" type="text/javascript"></script>
  </head>
  <body>
    <div id="data" gadget:type="http://www.ung-project.org/catalog.html">
      <div id="my_title">Thinkpad T400</div>
      <div id="my_price">400</div>
      <div id="my_price_currency">EUR</div>
    </div>
  </body>
</html>
```

The whole e-commerce site can thus be mirrored or cached using the same mirroring or caching tools as those which used back in 1993 with V6. UNG e-commerce application itself is implemented by so-called UNG gadgets which are described in a separate HTML file (http://www.ung-project.org/catalog.html) and rendered by the ung-engine.js. We are considering to extend this prototype to support mobile e-commerce by mirroring all HTML files of the catalog into the phone's localStorage.

4.   PROTOTYPE 3 : UNG ERP5

In UNG Docs, JIO is used to read and write HTML and SVG content from or to a simple storage such as WebDAV, localStorage or eventually remoteStorage. However, JIO could be used to read and write different types of content from/to complex applications such as ERP, CRM, CMS, Wiki, etc. We have derived from UNG Docs a prototype which shows that it is possible to list accounting transactions of an ERP in the same way we are listing office documents. To achieve this, we extended JIO so that the ERP application acts as a backend. On the front-end side, we are using UNG gadget framework to generate form layouts and display data.
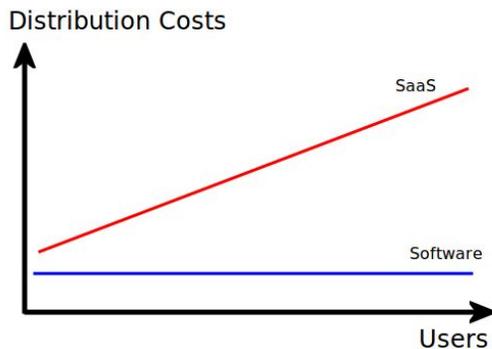

5.   ECONOMIC CONSIDERATION

The increasing adoption of Cloud and SaaS in particular changes the cost structure of software distribution by creating new hurdles towards the possibility of zero marginal cost.

Distribution costs for open source software are paid by the user who owns the PC which the software will be deployed on. The user also pays the internet access which the software has been downloaded with. The marginal cost of distribution of open source software is thus equal to zero for the author of the software, considering the possibility to use replication, proxies, content distribution networks and peer-to-peer to increase distribution bandwidth at no cost for the author.

Distribution costs of Software as a Service (SaaS) for the author of a software consists of the servers, electricity and network connectivity which must be purchased to provide the service. The more users, the more servers, the more electricity and the more network costs. The marginal cost of distribution of Software as a Service is no longer equal to zero, unless some kind of free infrastructure exists.

Software as a Service (SaaS) introduces a disruption in the economy of open source software. An independent author of open source software can become successful with SaaS only if he or she is able to pay the infrastructure required to serve millions of users. Unless infrastructure with zero marginal cost exists, independent open source authors will no longer be able to compete with large software publishers for any market where SaaS has become dominant, since part of the economic rationale of open source is based on the ability to gather of community of contributors at little or no cost.

W3C Widgets could help achieving again zero marginal cost for software distribution on the Cloud as long as storage has no cost. Zero cost storage is currently offered by different providers. Considering this opportunity, adopting W3C Widgets technology increasingly in the open source community should be a priority to protect open source competitiveness on the cloud since W3C Widgets offer the zero marginal cost property which if the foundation of open source economy.

6.   Future Work
The UNG project is part of the Resilience project, an industrial project supported by companies such as Nexedi, Alixen, Sagem Morpho, Wallix, XWiki, EADS Astrium, Alcatel Bell Lucent Labs, Institut Télécom and by research centers such as CEA, INRIA, Université de Paris 13. It is sponsored by French Ministry of Finance, Paris Regional Governement, Paris City and Essone Local Government.. As part of the project, UNG Docs will be extended to support more storage backends and more virtual storage. Planned storage backends include posgresql, Amazon S3, XWiki, Mioga and Unhosted remoteStorage. Virtual backends will support encryption, searchable encryption, content splitting and content enhancing.

Nexedi and Alixen will migrate their enterprise application software to the UNG framework within 2 years. It is expected through this migration to simplify the development of user interface independently of application backend and to reduce server load. EADS Astrium will experiment UNG framework to distribute high resolution sattelite images. Sagem Morpho will focus on privacy protection for sensitive content such as a citizen identity data.

W3C widgets will play a key role in this project to bridge the gap between web applications and native desktop applications. All results of the Resilience project will be provided in the form of W3C widgets.

The UNG project will also assess the possibility to deploy personal storage on mobile device such as tablets and phones. We plan to use SlapOS [9] decentralized cloud operating system to turn Android device into mobile cloud nodes. Initial experiments conducted in September 2011 show that Android is capable act as a WebDAV server or as a LAMP server. SlapOS relies on IPv6 to implement network neutrality on any device. The Resilience project has already developed a simple technology to provide IPv6 over HTTP to about any device and turn it into a server.

7.   Conclusion
The UNG Project shows that W3C Widgets are one of the best options for developers looking for a cost efficient approach to achieve resilience on the Cloud. W3C Widgets keep the javascript code and the CSS style locally while content is kept online on the Web as standard HTML5 which is eventually cached by proxies. Thanks to W3C widgets, authors of software can achieve the same level of ubiquity as Web based SaaS applications without having to pay for huge distribution costs of centralized Cloud infrastructure. Thanks to widespread adoption of mobile device such as phones and tablets, personal storage for W3C Widgets is available at zero cost for now and for the future.

8.   ACKNOWLEDGMENTS

9.   REFERENCES
[1] Bernard Lang and François Rouaix. The V6 Engine. Presented at the WWW5 Workshop: Programming the Web - in search for APIs. May 6, 1996. http://bat8.inria.fr/~lang/Papers/v6/paper.html

[2] TioLive - http://www.tiolive.com

[3] Salesforce - http://www.saleforce.com

[4] Google Docs - http://docs.google.com

[5] IWGCR - http://www.iwgcr.org

[6] Steve LOHR. Amazon's Trouble Raises Cloud Computing Doubts. The New York Times. April 22, 2011. http://www.nytimes.com/2011/04/23/technology/23cloud.html?_r=2

[7] Smets-Solanes, J-P; Atem de Carvalho, R (July–Aug 2003), "ERP5: a next-generation, open-source ERP architecture", IT Professional, 5, pp. 38–44, ISSN 1520-9202

[8] Uhosted. http://www.unhosted.org

[9] Jean-Paul Smets-Solanes, Christophe Cérin and Romain Courteaud. SlapOS: a Multi-purpose Distributed Cloud Operating System Based on an ERP Billing Model. IEEE 2011 International Workshop on Performance Aspects of Cloud and Service Virtualization. www.slapos.org