



Get Systems UA Ready

Blueprint for CIOs

1 June 2017



TABLE OF CONTENTS

Purpose of this Blueprint	3
Why you should care	3
Background on Universal Acceptance	3
Universal Acceptance and the Internet Industry	3
Steps toward UA Readiness	4
Logging	4
Inventory	4
Control	4
Evaluate potential need for UA Remediation	5
Electronic Data Interchange and other industry specific transfer details	5
Speed & Timeline	5
Barriers to Success	6
Test Suites	6
Appendix 1 – Architecture Decisions	7
Appendix 2 – Internet Industry Software Status	8
Webserver	8
E-Mail / Groupware	8
Email Clients / Mobile Apps	9
Nameserver	10
Accounting, HR, Payroll	10
Routing and Switching Management	10
Customer Logistic/Fulfillment	10
Domain Management	10
Spam-Filter	10
Firewall	11
Control Panels	11



■ Purpose of this Blueprint

This document is a generic guideline for Chief Information Officers (CIOs) to use when reviewing their software systems to ensure that they are able to accept all domain names and email addresses. The Universal Acceptance Steering Group (UASG) calls this being 'UA Ready'.

Why you should care

The Internet, and Internet Standards, are constantly evolving. In 2010 there were a number of new top-level-domains introduced in non-ASCII scripts. In 2013 the Root Zone saw the start of an introduction of more than 1,000 additional top level domains - in ASCII and non-ASCII scripts. These new top-level-domain names give people choice and ensures competition in the domain name world. No longer can applications assume that the top-level domain name space will be just two or three characters or just in ASCII or even static.

In addition, email mailbox names can contain non-ASCII characters through the Email Address Internationalisation (EAI) standards.

There are literally billions of people that use languages based on non-Latin alphabets who have started being able to use the Internet to its fullest extent in their mother tongue. Much of the next billion users will come from regions with non-ASCII languages.

Your systems need to be brought up to speed with today's Domain Name System (DNS), which is diverse and multi-lingual. Being UA Ready is probably the key to the biggest business opportunity you're missing.

■ Background on Universal Acceptance

Universal Acceptance (UA) is the state where all valid domain names and email addresses are accepted, validated, stored, processed and displayed correctly and consistently by all applications, devices and systems.

Due to the rapidly changing domain name landscape, many systems do not recognize or appropriately process new domain names, primarily because the Top-Level Domain (TLD) may be more than three characters in length or in a non-ASCII format (Internationalized Domain Names, or IDNs). New standards in email (EAI – Email Address Internationalization) have also been introduced to accommodate the non-ASCII domain name space as well as Unicode in the mailbox name.

Not all online portals are primed for the opening of a user account with one of these new email addresses. While filling out online forms, top-level domains that exceed the previous standard length of two or three characters and e-mail addresses that are based on Unicode are not always accepted. To give you one example: Over 90% of all websites tested accept our ASCII@new four-character-TLD, but less than 5% accept our unicode@idn.idn!

The UASG, supported by the Internet Corporation for Assigned Names and Numbers (ICANN), is a community-led initiative working on creating awareness and identifying and resolving problems associated with Universal Acceptance. The purpose of these efforts is to help ensure a consistent and positive experience for Internet users globally.

■ Universal Acceptance and the Internet Industry



The Internet Industry consists of Internet, Cloud and E-Mail Service Providers as well as domain name registries and registrars, hosting providers and consultants. Together they provide the cornerstones of the Internet Industry. Every email, every request to display a website passes your systems on the application level. This is why the UASG is encouraging you to get all of your systems UA Ready and we seek your support to accomplish the mission to get every domain name working everywhere with every software system and website.

This document is aimed specifically at the Internet Industry:

- ▶ Internet Service Providers (ISPs)
- ▶ Cloud Service Providers
- ▶ Hosting Companies
- ▶ Email Service Providers (ESPs)
- ▶ Domain Registries
- ▶ Domain Registrars
- ▶ Consultants and resellers

The goal of this document is to provide the IT teams in the Internet Industry a starting point on their own quest for UA Readiness.

This document also keeps track of the UA Readiness of software commonly used within the Internet Industry. The Appendices provide a list and the state of UA readiness.

■ **Steps toward UA Readiness**

Logging

A big first step would be to create a clearer picture of the status quo: Does your company encounter UA Issues? How many UA Issues do you, your engineers and your support staff receive per day, week or month?

Together with ICANN, the UASG offers you direct support via an online issue logging system: <https://uasg.tech/global-support-centre>

Inventory

The first steps you will want to go through are to create an inventory of all the applications under your responsibility. All IT operations should have this readily to hand, but realistically that is probably not the case.

You can gather this information from:

- ▶ Disaster Recovery Plans which will have a list of all applications and tools and vendors for each.
- ▶ Checking with your development staff, especially when you are running in-house developed software, applications and Application Programming Interfaces (APIs) or in-house developed modifications and extensions of standard software.
- ▶ Checking with your Systems Architect or the owner of your Data Dictionary. Indeed, a good data dictionary will make the assessment relatively easy.
- ▶ Checking with your procurement team who may have contracts for all software or internal billing systems that you may use.

Control



Once you have your list of applications that you support (and ideally you will support all your organizations' applications, but if you don't then you will need to involve the business units that control their own applications) do some analysis of each application:

Is the software in scope?	Who is the customer?	What level of control do you have?
<ul style="list-style-type: none">▶ Does it have a field for an email address?▶ Does it have a field for a domain name?	<ul style="list-style-type: none">▶ Internal▶ Customer Facing▶ Supplier Facing▶ Industry engagement	<ul style="list-style-type: none">▶ Source Code and full rights▶ Strong relationship with the supplier▶ A relatively small customer

This analysis will determine the approach that you take and the priority you give to each application.

Evaluate potential need for UA Remediation

If an application, such as Fixed Asset Register, does not reference a domain name or email address, then there is no need to evaluate it for UA Readiness.

If the application does include these, then you need to determine the ease of remediation and the priority.

Applications for which you have the source code and skills will be easier to address. Where you (or your collective industry) have strong relationship with a supplier may also be relatively easy to place a change request that will be processed within a short period of time. Where you are a 'small fish in a big pond', you may need to find out where UA Readiness is in your supplier's development roadmap.

A growing number of open source programming languages and utilities are being made UA Ready. This could mean that recompiling in the latest version may be sufficient to make your systems UA ready.

Electronic Data Interchange and other industry specific transfer details

They key issue when exchanging data electronically with outside parties is to make sure that the expectations are explicitly documented.

Some legacy data exchange facilities may be limited to 7 bit character sets. More recent implementations will support 8 bit.

With the Extensible Provisioning Protocol (EPP), for example, while the protocol does support 8 bit character sets, it is recommended that the specifics to be carried in each field should be explicitly agreed between parties.

Speed & Timeline

The nature of Universal Acceptance is such that there is not likely to be a specific project to get applications UA Ready.

Instead, there are a couple of approaches:



- ▶ Adjusting validation routines to allow for long ASCII TLDs is a useful and relatively easy first step.
- ▶ UA should be built into the IT Road Map and should be applied anytime an application is opened for routine maintenance and enhancements.
- ▶ The priority for application enhancement will depend on whether there are pressing demand from customers, suppliers or business partners.

Barriers to Success

The biggest barrier to success is the reliance of third party application providers where you are NOT a 'big fish'.

Not having an existing inventory of applications and not having a current data dictionary can also be barriers. However, creation of these can be positive side benefits of a initiative to achieve UA Readiness.

Test Suites

A number of live and valid domain names and email addresses are listed in the document [UASG004 "Use-Cases-for-UA-Readiness-Evaluation"](#) provided by the UASG. These can be used for testing without the need to register your own. Even semi-automated tool is available on <https://eai.xgenplus.com> for testing email servers and its deliverability.



■ Appendix 1 – Architecture Decisions

When reviewing your application and, indeed, your entire software eco-system, there may be some architecture issues that you need to consider. Once decided, then consistency across applications will make things easier going forward.

The UASG recommends using 8 bit characters through the applications, doing Punycode conversions only when sending data across networks. Where users may not be reasonably familiar with multiple scripts, displaying in both Unicode and punycode may be useful.

- ▶ Is your text stored in 7bit or 8bit character sets? Seven bit character sets limit your applications to ASCII codes, where 8 bit character sets allow a full range of Unicode characters, which is required to store Unicode-based domain names and email addresses.
- ▶ How are you storing domain names? Are you storing them as punycode 7 bit character sets (which will require conversion for every input and output process) or are you storing them as 8 bit character set – native Unicode.
- ▶ Depending on how you store text, domain names and email addresses will also affect your search and sort routines.
- ▶ What is your approach to validation?
 - Do you validate against a table? If so, then the table must be kept up to date because top level domain names are changing on an almost daily basis.
 - Do you validate against the Domain Name System (DNS)?
 - Do you validate against the heuristic characteristics of a domain name or email address (e.g., validate domain extensions by length)?
- ▶ How are you going to display the data?
 - Are domain names displayed in Unicode – which are readable and understandable to native script readers or do you display in Punycode or do you display both?



Appendix 2 – Internet Industry Software Status

Note: This is a dynamic list and if you identify applications that should be added, please send suggestions to info@uasg.tech

Application/Website	Contacted	Status	Last Update	Comments
Webserver				
Odin / Parallels / Confixx				
Citrix				
Apache				
Nginx				
Lightspeed				
E-Mail / Groupware				
Open Xchange / dovecot				
CommuniGate				
IBM / Lotus Notes				
Novell Groupwise				
Sun ONE				
Microsoft Exchange				
Citadel/UX				
Collanos Workplace				
Microsoft-SharePoint				
Zimbra Collaboration Suite (ZCS)				
Roundcube				
Squirrelmail				
Comindware Tracker				
FirstClass				



Kerio Connect				
Kolab				
HORDE Groupware				
EGroupware				
phpGroupWare				
GROUP-E				
PHProjekt				
Tine 2.0				
Tiki – CMS Wiki Groupware				
Zarafa				
SendGrid				
powerMTA / port25/ message systems				
AWS Amazon Simple Email Service (SES)				
xworks / mediabeam				
Postfix				
Exim				
Sendmail				
XgenPlus				
Email Clients / Mobile Apps				
Microsoft Outlook 2016 (windows)				
BharatSync Communicator (Android)				
Outlook (android/ios/windows)				
Gmail (android/ios/windows)				
XgenMail (IOS)				



EAI (IOS)				
Nameserver				
PowerDNS Server				
BIND (Berkeley Internet Name Domain)				
Djbdns				
Dnsmasq				
Knot DNS (cz.nic)				
Microsoft Windows DNS				
NSD (Name Server Daemon)				
Unbound				
Namecoin				
Accounting, HR, Payroll				
DATEV				
QuickBooks Online				
Xero				
Routing and Switching Management				
Customer Logistic/Fulfillment				
Domain Management				
AUTODNS (InterNetX)				
ironDNS				
PartnerGate				
RRPproxy				
Spam-Filter				
Spamassasin				



d9t				
Mailcleaner				
Barracuda				
Livestep				
Edgewave				
FuseMail				
Mimecast				
Proofpoint				
SpamWeeder				
MailCleaner				
HornetSecurity				
Netintelligence				
Panda Security				
Themailaundry				
Spamjadoo				
Firewall				
Watchguard				
f5				
Securepoint Security Solution				
ConfigServer				
Control Panels				
cPanel				
Plesk				



DirectAdmin				
Error Checking on Subscribers				
Maxmind.com				
Secure Domain Foundation				