RFC3629

Network Working Group F. Yergeau
Request for Comments: 3629 Alis Technologies
STD: 63 November 2003
Obsoletes: 2279
Category: Standards Track

UTF-8, a transformation format of ISO 10646

Status of this Memo

Copyright Notice

Abstract

ISO/IEC 10646-1 defines a large character set called the Universal Character Set (UCS) which encompasses most of the world's writing systems. The originally proposed encodings of the UCS, however, were not compatible with many current applications and protocols, and this has led to the development of UTF-8, the object of this memo. UTF-8 has the characteristic of preserving the full US-ASCII range, providing compatibility with file systems, parsers and other software that rely on US-ASCII values but are transparent to other values. This memo obsoletes and replaces RFC 2279.

Table of Contents

1. Introduction

ISO/IEC 10646 [ISO.10646] defines a large character set called the

Universal Character Set (UCS), which encompasses most of the world's writing systems. The same set of characters is defined by the Unicode standard [UNICODE], which further defines additional character properties and other application details of great interest to implementers. Up to the present time, changes in Unicode and amendments and additions to ISO/IEC 10646 have tracked each other, so that the character repertoires and code point assignments have remained in sync. The relevant standardization committees have committed to maintain this very useful synchronism.

ISO/IEC 10646 and Unicode define several encoding forms of their common repertoire: UTF-8, UCS-2, UTF-16, UCS-4 and UTF-32. In an encoding form, each character is represented as one or more encoding units. All standard UCS encoding forms except UTF-8 have an encoding unit larger than one octet, making them hard to use in many current applications and protocols that assume 8 or even 7 bit characters.

UTF-8, the object of this memo, has a one-octet encoding unit. It uses all bits of an octet, but has the quality of preserving the full US-ASCII [US-ASCII] range: US-ASCII characters are encoded in one octet having the normal US-ASCII value, and any octet with such a value can only stand for a US-ASCII character, and nothing else.

UTF-8 encodes UCS characters as a varying number of octets, where the number of octets, and the value of each, depend on the integer value assigned to the character in ISO/IEC 10646 (the character number, a.k.a. code position, code point or Unicode scalar value). This encoding form has the following characteristics (all values are in hexadecimal):

- Character numbers from U+0000 to U+007F (US-ASCII repertoire) correspond to octets 00 to 7F (7 bit US-ASCII values). A direct consequence is that a plain ASCII string is also a valid UTF-8 string.

- US-ASCII octet values do not appear otherwise in a UTF-8 encoded character stream. This provides compatibility with file systems or other software (e.g., the printf() function in C libraries) that parse based on US-ASCII values but are transparent to other values.

- Round-trip conversion is easy between UTF-8 and other encoding forms.

- The first octet of a multi-octet sequence indicates the number of octets in the sequence.

- The octet values C0, C1, F5 to FF never appear.

- Character boundaries are easily found from anywhere in an octet stream.

- The byte-value lexicographic sorting order of UTF-8 strings is the same as if ordered by character numbers. Of course this is of limited interest since a sort order based on character numbers is almost never culturally valid.

- The Boyer-Moore fast search algorithm can be used with UTF-8 data.

- UTF-8 strings can be fairly reliably recognized as such by a simple algorithm, i.e., the probability that a string of characters in any other encoding appears as valid UTF-8 is low, diminishing with increasing string length.

UTF-8 was devised in September 1992 by Ken Thompson, guided by design criteria specified by Rob Pike,

rfc1901.pdf 10-28
Introduction to Community-based SNMPv2

中文版RFC文档 01-09
来自以下
组织：中国互动出版网（http://www.china-pub.com/）
RFC文档中文翻译计划（htt...

字符编码 UTF-8 学习笔记_rfc3629 utf-8_kfepiza的博客 7-24
一个字节二进制有几个1开头, 后面就有几减一个10开头的字节 五字节和六字节不是Unicode编码范围, 20...

UTF-8 8-bit Unicode Transformation Format 万国码_一秒变桌子的博客... 8-12
UTF-8(8-bitUnicodeTransformation Format)是一种针对Unicode的可变长度字符编码,又称万国码,由Ken ...

RFC协议标准 11-17
vlan/acl/qos/lacp/802.1x/802.3x/路由、生成树协议、组播协议等标准规范

RFC 3561 aodv 11-06
This memo defines an Experimental Protocol for the Internet community. It does not specify an Internet...

C#高性能大容量SOCKET并发(七):协议字符集_SQLDebug_Fan的博客 7-16
UTF-8是UNICODE的一种变长字符编码又称万国码,由Ken Thompson于1992年创建。现在已经标准化为R...

IOT-MQTT协议-简介_leeahuamsg的博客 7-30
Bradner,S。,“用于RFC指示需求水平的关键词”,BCP 14,RFC 2119,1997年3月.http://www.ietf.org/rfc/rfc2...

rfc791文档 11-30
rfc791，里面写了很多规范，向深入学习网络知识的同学可以下载来看看

编码知识学习笔记之一 2023跟着小虎玩着去软考 1688
编码知识学习笔记之一 一．有哪些编码 1. ANSI 2.Unicode 3.Unicode big Endian 4.Unicode - ASCII Esca...

物联网传输协议MQTT研究_草根大哥的博客 8-4
在一个UTF-8字符数据编码的字符串必须是格式良好的UTF-8的Unicode规范[UNICDOE]定义和重述的RFC...

编码相关的问题_zhengudaoer的博客 7-30
IETF的RFC2781和RFC3629以RFC的一贯风格,清晰、明快又不失严谨地描述了UTF-16和UTF-8的编码方...

UTF-8 IT老兵的驿站 1366
摘自：https://zh.wikipedia.org/wiki/UTF-8，个别地方有几个特殊符号没被处理，显示成层叠状态。 维...

UTF-8编码表 11-17
UTF，是UnicodeTransformation Format的缩写，意为Unicode转换格式。UTF-8是UNICODE的一种变长...

MQTT V3.1.1 协议 规范 Xanthium 3242
目录 1.简介 1.1术语 1.2 数据表示 1.2.1 位 1.2.2整数数据值 1.2.3 UTF-8编码的字符串 2 MQTT控制包格...

下载的附件名总乱码？你该去读一下 RFC 文档了！ Java笔记虾 643
纸上得来终觉浅，绝知此事要躬行Web 开发过程中，相信大家都遇到过附件下载的场景，其中，各浏览...

rfc2279utf8协议 09-09
ISO/IEC 10646-1 [ISO-10646]定义了一种多8比特字节字符集，称作通用字符集（UCS）， 它包含了世界...

精述字符编码（读这篇就够了） Dablelv 的博客专栏。 6942
UCS（Universal Character Set，通用字符集）是由ISO制定的。

RFC6020 - YANG语言标准中文 dolphin98629的专栏 1799
RFC6020 - YANG语言标准中文 2016年08月05日 14:49:04 阅读数：12297 YANG - A Data Modeling Lan...

SSH中的安全 | 从SSH协议看身份验证底层原理 qq_41909850的博客 282
本文介绍了 SSH 协议在验证用户身份过程中的实现细节，想帮助读者更加深入的了解 SSH 客户端与服...

websocket规范 RFC6455 中文版 热门推荐 Stoneson 2万+
翻译自： http://tools.ietf.org/rfc/rfc6455.txt InternetEngineering Task Force (IETF) I. Fette...

中文RFC文档资源 weixin_34148456的博客 194
[URL=http://www.infoxa.com/RFC/rfc/RFC1.txt]RFC1 主机软件[/URL]RFC2 主机软件[URL=http://www.i...

SIP RFC 3261 中文文档（RFC3261） 风中骄子 7912
7、SIP消息：SIP协议是一个基于文本的协议，使用UTF-8字符集（RFC2279[7]）。一个SIP消息既可以...

utf8编解码详解 twwk120120的专栏 4522
utf8编解码详解及简单应用 编码规则 UTF-8是Unicode的一种实现，是一种变长字节编码方式。对于某一...

utf-8编码算法 hongweigg的专栏 4116
unicode字符集是我们世界上最完善最全面的字符集，几乎包含了世界上所有的字符。其实可以这么理...

abap 执行rfc 最新发布 08-03
在ABAP中，可以通过执行RFC（远程函数调用）与远程系统进行通信。ABAP可以充当RFC客户端或者R...

“相关推荐”对你有帮助么？

非常没帮助 没帮助 一般 有帮助 非常有帮助

lyclowlevel

码龄19年 暂无认证

46	102万+	69万+	13万+	
原创	周排名	总排名	访问	等级

1894	9	15	11	30
积分	粉丝	获赞	评论	收藏

私信 关注

搜博主文章

热门文章

分类专栏

最新评论

单精度浮点数（IEEE754）

lvhuatan: 瞎鸡儿说

单精度浮点数（IEEE754）

Joyhooian: little endian是低地址存放低有效字节，所以42 F7 00 00 应该是 00 00 4...

你的“重叠IO”是真正异步的吗？

lyl00982: 请问，我在一个while（1）循环里，循环写入，写着写着就不是异步IO了...

你的“重叠IO”是真正异步的吗？

小小胡孙: 不错，对异步IO和重叠IO理解更深了。

z-order引出的问题

lyclowlevel 回复 这里指的并存有歧义。使用ws_child|ws_popup是可以正常创建窗口...

您愿意向朋友推荐“博客详情页”吗？

强烈不推荐 不推荐 一般般 推荐 强烈推荐

最新文章

2012年 3篇	2011年 17篇
2010年 23篇	2009年 6篇
2007年 5篇	