Avatarx

博客园 首页 新随笔 联系 管理 订阅 🚟

调用QString::arg()一次:

随笔-1695 文章-0 评论-3

Qt 格式化字符串

复制代码

Qt字符串格式化性能比较

Qt字符串格式方法有三种, QString::arg(), QString::sprinft()和QStringList::join(). 今天我做了个简单的性能测试, 希望对各位有所帮助.

```
1. QString s("1234567890");

// 开始计时
for (int i = 0; i < 10000; ++i) {
    QString str("%1");
    str.arg(s);
}
// 停止计时
```

```
调用QString::arg()十次:
复制代码
1. QString s("1234567890");
```

// 停止计时

```
// 开始计时
for (int i = 0; i < 10000; ++i) {
    QString str("%1%2%3%4%5%6%7%8%9%10");
    str
        .arg(s)
        .arg(s)
```

```
调用QString::sprinft()一次:
复制代码
1. char s2[] = {"1234567890"};

// 开始计时
for (int i = 0; i < times; ++i) {
    QString().sprintf("%d", s2);
    }
    // 停止计时
```

昵称: Avatarx园龄: 5年10个月粉丝: 198关注: 10+加关注

<	2019年12月					>
日	_	=	Ξ	四	五	<u>, , , , , , , , , , , , , , , , , , , </u>
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4
5	6	7	8	9	10	11

搜索

找找看
谷歌搜索

常用链接

我的随笔 我的评论 我的参与 最新评论 我的标签

我的标签

```
qt mvc(4)
qt pro(2)
qt python(1)
qt QSettings(1)
qt qt5 基础(1)
qt sql 数据库操作(1)
qt stylesheet 美化 控件样式(1)
qt 编译oracle驱动 oci(1)
qt 程序插件(1)
qt 程序居中显示(1)
更多
```

随笔分类

```
线程的同步与异步
.Net Core(3)
ACE(2)
AI(人工智能)(73)
Android
ARM(28)
boost库(30)
C#(154)
c++(88)
C++11(34)
codeblocks(2)
cowboy
CppCheck(1)
```

C语言(53)

```
DeepIn(1)
                                                                                      DirectUI
                                                                                      Dos(4)
调用QString::sprinft()十次:
                                                                                      D语言(3)
复制代码
                                                                                      EF(3)
  1. char s2[] = {"1234567890"};
                                                                                      ElasticSearch(1)
                                                                                      ftp(20)
                                                                                      GO语言(46)
       // 开始计时
                                                                                      Gtk+
       for (int i = 0; i < times; ++i) {
                                                                                      Gun Plot(5)
       Honker修成之路(2)
                                                                                      IOS(3)
       // 停止计时
                                                                                      IPC(1)
                                                                                      iscsi(3)
                                                                                      Java(58)
                                                                                      Json(3)
                                                                                      JSP(5)
调用QStringList::join()一次:
                                                                                      LabView(30)
复制代码
                                                                                      LING(1)
  1. QString s("1234567890");
                                                                                      linux(191)
                                                                                      linux mingw(4)
                                                                                      Linux驱动开发(20)
       // 开始计时
                                                                                      lua(12)
       for (int i = 0; i < times; ++i) {
                                                                                      MatLab(1)
       QStringList strList;
                                                                                      metro(11)
       strList.append(s);
                                                                                      MFC(69)
       strList.join(QString());
                                                                                      miniGui(1)
                                                                                      MV(3)
       }
                                                                                      mysql(77)
       // 停止计时
                                                                                      ODB(14)
                                                                                      OLEDB(12)
                                                                                      OpenCv(5)
                                                                                      OpenMp(4)
                                                                                      PostgreSQL(3)
                                                                                      pvthon(14)
调用QStringList::join()十次:
                                                                                      QML(16)
复制代码
                                                                                      QQuick(17)
  1. QString s("1234567890");
                                                                                      qt(227)
                                                                                      qt layout
                                                                                      qt mvc(21)
       // 开始计时
                                                                                      qt oracle(37)
       for (int i = 0; i < times; ++i) {
                                                                                      qt 打包发布(9)
       QStringList strList;
                                                                                      Qt 底层(3)
       strList.append(s);
                                                                                      Qt 国际化(6)
       strList.append(s);
                                                                                      Qt 界面特效(12)
       strList.append(s);
                                                                                      qt 皮肤 (qss) (11)
       strList.append(s);
                                                                                      qt 三维绘图(7)
       strList.append(s);
                                                                                      qt 信号和槽(7)
       strList.append(s);
                                                                                      Ot 疑难杂症(2)
       strList.append(s);
                                                                                      qt_3D(7)
                                                                                      qt_Container(1)
       strList.append(s);
                                                                                      qt_dll(7)
       strList.append(s);
                                                                                      qt_drag
       strList.append(s);
                                                                                      qt_draw(1)
       strList.join(QString());
                                                                                      qt_encryption
                                                                                      qt_event(10)
       // 停止计时
                                                                                      at excel(6)
                                                                                      qt_gui(46)
                                                                                      Qt_HTML5(2)
                                                                                      qt_IO(4)
                                                                                      qt_json(2)
测试结果:
                                                                                      at network(9)
  运行一次的耗时: QString::arg() 0.412纳秒 < QStringList::join() 0.625纳秒 < QString::sprinft()
                                                                                      qt_sql(18)
  运行十次的耗时: QStringList::join() 2.426纳秒 < QString::arg() 5.175纳秒 < QString::sprinft()
                                                                                      qt_thread(25)
                                                                                      qt_XML(13)
                                                                                      qt_串口(5)
现在让我们来看看这些函数是怎么完成各自的任务的.
                                                                                      qt_动画(1)
                                                                                      Ot 新闻(1)
                                                                                      qt_语音(2)
  先来看看QString::arg(). 这个函数的实现非常简单,它先将传入的参数转换成QString格式的字符串参数,
                                                                                      QtException
换源字符串里的替换符(比如%1, %2, %3什么的). replaceArgEscapes首先计算一下新的字符串将会有多
                                                                                      QtScript(2)
再将源字符串和字符串参数合并到临时字符串中. 一次操作就这么完成了.
                                                                                      qt-vlc(8)
  在这个过程中,比较消耗时间的是创建了一个临时字符串. 随着累加调用QString::arg(), 当调用十次的时间
                                                                                      qt测试(4)
秒是消耗在创建10个临时字符串当中. 还有重复多次的内存拷贝. 所以QString::arg()累加调用的越多, 它的执
                                                                                      qwt(8)
                                                                                      Redis (21)
```

Scade(2)

下面再来看看QString::sprintf(),这个函数是仿照C函数的sprintf(),使用方法上一致,这个函数也不复杂,QString::vsprintf()中先创建个临时字符串,然后类似于replaceArgEscapes的方式循环追加到这个临时字符串的大小,调用的是QString::append().这个函数在缓存够大的时候直接追加数据,如果不够大则重新分配足QString::arg()本质上并没有太大区别.因为每次都会生成个新的字符串,并将数据拷贝进去.

好的东西都要放到最后面再说,现在我们来看看QStringList::join()这个方法,这个方法为什么合并10个数明了.它先循环获取每个字符串的长度,这样就可以计算出整个字符串的长度,注意,是整个,不是一部分.实际.效率永远都是O(1).

那么说到这里,QString::arg()和QString::sprintf()的方法基本一样,为什么差距那么大呢? 嘿嘿,你们有没得QString(),而给QString::sprintf传递的是char*,所以arg执行的时候不需要再把参数转换成字符串参数了,QString::sprintf不得不将char*类型的字符串转换成为QString().这也是没办法的事儿,因为QString::spri

现在我把QString::arg()和QString::join()里面预先构建的QString()字符串都替换成临时构建. 让我们来看得

运行一次的耗时: QString::arg() 0.642纳秒 < QStringList::join() 0.792纳秒 < QString::sprinft() 运行十次的耗时: QStringList::join() 4.363纳秒 < QString::arg() 7.171纳秒 < QString::sprinft()

这次算是公平的了,从上面的数据可以看出来,为什么官方不推荐使用QString::sprintf()这个函数了,首先是这C/C++程序员的使用习惯. 基本上可以说一无是处了. 那么为啥也没有推荐使用QStringList.join()呢? 主要是father %1 has two bros, %2 was killed by malaria, %3 is still alive.", 如果用QString::arg()是如此成这样: "My father " << "Old John" << " has two bros, " << "one" << " was killed by malaria, " 替换三个, 一个是合并7个.

我希望能通过这次测试各位从事Qt的朋友们,QString::sprintf()完全可以放弃了,认为这个性能会如何如何好函数;剩下99%的时候都应该使用QString::arg(),因为各位不会一下子格式化十个八个参数的,一般都是一下个则要果断选择QStringList::join()了.

写了这么多,希望能对各位有所帮助.

补充:

非常谢谢<u>dbzhang800" data-card-url="pw_ajax.php?action=smallcard&type=showcard&uid="tarccheckUrl(this)" id="url_1">dbzhang800</u>的补充,QString::arg()的模板函数版本平时真的没怎么用过,哈>=10的参数,所以把十个参数改成九个参数进行比较.

下面是最新的性能比较结果.

运行

运行十次的耗时: QStringBuilder 2.126纳秒 < QStringList::join() 3.781纳秒 < QString::arg() 模协 纳秒 < QString::sprinft() 8.933纳秒;

从上面的运行结果来看,QString::arg()模板版本只是略微强于QString::arg()的多次调用版本. 函数内部仍然式仍然面临多次分配内存的操作. 而且QString::arg()模板版本其实并不是模板函数,只是一次接受多个QStrir它已经丢失了QString::arg()普通版本的一大优势--基本类型通杀. 也就是说,如果你要使用QString::arg()的据都自己转换成QString类型的,否则你就没办法使用.

牛叉的总要放在后面说,是吧. dbzhang800" data-card-url="pw_ajax.php?action=smallcard&type:onclick="return checkUrl(this)" id="url_2">dbzhang800翻译的文章中已经对这个有详细的介绍了. 使用个是内建的,然后注意字符串连接时候的连接符是%,而不是+.

再次感谢dbzhang800" data-card-url="pw_ajax.php?action=smallcard&type=showcard&uid=" tarceteckUrl(this)" id="url_4">dbzhang800的帮助,希望大家能够继续补充!

dbzhang800

呵呵,写的真不短啊。我补充一点。

都是QString的话,如果用 arg的话,应该选择:

QString("%1 %2 %3 %4").arg(s1, s2, s3, s4)

而不是

QString("%1 %2 %3 %4").arg(s1).arg(s2).arg(s3).arg(s4)

shell脚本(1) sonar(编码质量控制) SqlLite(16) SQLServer(106) TFS(1) TTB(3)unix(18) VS(38) VxWorks(6) WCF(18) Win32(65) windows(7) Windows驱动开发(1) wireshark(18) WPF(61) 测试(5) 禅道(5) 程序版本管理(20) 大数据(2) 单片机(20) 多核编程(7) 帆樯(4) 工程管理(Project)(1) 工程检测(2) 工程建模(5) 汇编(14) 机器人(2) 计算机安全技术(2) 加密技术(5) 媒体编辑(3) 模糊逻辑(1) 内存管理(7) 区块链(21) 人工智能(8) 软件缺陷追踪(4) 设计模式(7) 数学 (高数,线性代数,概率论) (1) 网络(1) 网络爬虫(2) 网站建设(2) 问题与总结(4) 无锁编程 (LockFree) (6) 系统安装(30) 系统权限管理(6)

杀统仪帐官理(€

虚拟技术(4)

硬件自检(1)

远程技术(9)

厶

自动化测试

随笔档案

2019年11月(27)

2019年10月(18)

2019年9月(13) 2019年8月(37)

2019年7月(37)

2019年5月(3)

2019年4月(2)

2019年3月(14)

2019年2月(1)

2018年11月(28) 2018年10月(31)

2018年9月(14)

010年2月(17)

2018年8月(17)

2018年6月(3)

2018年5月(6)

2018年4月(1)

(018年4月(1)

2018年3月(16)

2018年1月(6)

2017年12月(12)

2017年11月(15) 2017年10月(11)

2017年10万(11 2017年9月(13)

2017年3月(13)

2017年7月(29)

另外字符串链接的话: s1 + s2 + s3 + s4 也是比较常用的,但性能就不如 s1 % s2 % s3 % s4 这种写法了

分类: <u>qt</u>





<u>Avatarx</u> <u>关注 - 10</u> <u>粉丝 - 198</u>

± - 10 <u>½ - 198</u>

+加关注

« 上一篇: 实现C++模板类头文件和实现文件分离的方法 » 下一篇: 搭建Windows下Java Web开发环境

posted @ 2015-02-16 10:20 Avatarx 阅读(15689) 评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

0

(评论功能已被禁用)

【推荐】超50万行VC++源码:大型组态工控、电力仿真CAD与GIS源码库

【推荐】腾讯云海外1核2G云服务器低至2折,半价续费券限量免费领取!

【推荐】阿里云双11返场来袭,热门产品低至一折等你来抢!

【推荐】天翼云双十一翼降到底,云主机11.11元起,抽奖送大礼

【推荐】流程自动化专家UiBot,体系化教程成就高薪RPA工程师

【活动】京东云服务器_云主机低于1折,低价高性能产品备战双11

【优惠】七牛云采购嘉年华,云存储、CDN等云产品低至1折



相关博文:

- 趣味算法:字符串反转的N种方法
- · QT 格式化字符串功能
- ·Java字符串之性能优化
- · Qtsprintf_s函数格式化字符串出错
- ·Java字符串之性能优化
- » 更多推荐...

深度回顾! 30篇好文,解析历年双十一背后的阿里技术秘籍



最新 IT 新闻:

- · 李洪元回应华为声明: 大家看看先, 我听全国人民的
- · 逆康普顿散射打造"史上最强的光"
- ·暴风集团: 高管已全部离职, 公司仅剩10余人
- 引力波天文学开启黑洞研究新纪元
- ·金山云拟发行约5509万股D+系列优先股
- » 更多新闻...

2016年10月(2) 2016年9月(1) 2016年8月(1) 2016年7月(14) 2016年6月(71) 2016年5月(63) 2016年4月(76) 2016年3月(43) 2016年2月(1) 2016年1月(9) 2015年12月(1) 2015年11月(31) 2015年10月(30) 2015年9月(87) 2015年8月(10) 2015年7月(12) 2015年6月(7) 2015年5月(97) 2015年4月(52) 2015年3月(53) 2015年2月(7) 2015年1月(49) 2014年12月(61) 2014年11月(58) 2014年10月(34) 2014年9月(5) 2014年8月(15) 2014年7月(41) 2014年6月(68) 2014年5月(144)

2017年6月(12)

2017年5月(41)

2017年4月(36) 2017年3月(56)

2017年2月(18)

2017年1月(10)

2016年12月(8) 2016年11月(5)

最新评论

1. Re:LabView培训

链接没有了楼主

2014年4月(4)

--梦游城市

2. Re:Qt 类外调用一个 private slots 函数

鬼斧神工的做法,见识了。谢谢。谢谢你那 么多有水平的文章。

--tadpole999

3. Re:qt QThread

你好,我最近也在弄qt线程,但是会发现一个问题,就是正常运行都没问题,但是如果 线程执行过程中,用户强制退出程序的时 候,此时进行析构的时候会出错,这怎么解 决啊?

--yzfree

阅读排行榜

- 1. 详细的OS X Yosemite 10.10懒人版安 装教程(29723)
- 2. wireshark如何抓取本机包(29554)
- 3. c# sleep 例子(25570)
- 4. qt超强精美绘图控件 QCustomPlot— 览及安装使用教程(22979)
- 5. 5 个免费的受欢迎的 SQLite 管理工具 (21341)

评论排行榜

- 1. qt QThread(1)
- 2. Qt 类外调用一个 private slots 函数 (1)
- 3. LabView培训(1)

推荐排行榜

- 1. wireshark如何抓取本机包(7)
- 2. C# volatile 关键字(4)
- 3. EF中使用SQL语句或存储过程(2)
- 4. Linux驱动开发学习的一些必要步骤(2)
- 5. udp 不需要 listen(2)

Copyright © 2019 Avatarx
Powered by .NET Core 3.0.1 on Linux