Smuxi Issues [FROZEN ARCHIVE] - Task # 387: Use own hashing algorithm for nick colors

Status:	Closed	**Priority:**	Normal
Author:	Mirco Bauer	**Category:**	Engine
Created:	06/14/2010	**Assigned to:**	Mirco Bauer
Updated:	08/15/2010	**Due date:**	
Complexity:	Medium		
Subject:	Use own hashing algorithm for nick colors		
Description:	Currently Smuxi relies on the String.GetHashCode() implementation of the runtime. This has the drawback that the value could change over time (between runtime versions and implementations) leading to suddenly different colors. Thus Smuxi should use an own algorithm to guarantee the same nick colors over time. See ProtocolManagerBase.GetIdentityNameColor()		

Associated revisions

08/15/2010 06:02 PM - Mirco Bauer

Implemented new message builder API

Implemented a new message builder API which allows to create messages with

a much nicer API and using less code.

The new API generates more distinct nick-colors by using a color combination

consisting of colored tags (<>) and colored nick name. (closes: #388)

The own nick is now bold to make it more distinct from other nick names.

(closes: #419)

Refactored Engine-IRC and Engine-Twitter to make use of the new API.

Imported MIT/X11 licensed CRC32 implementation for stable hashcodes needed for

the unique nick-colors feature. (closes: #387)

History

06/27/2010 11:24 AM - Mirco Bauer

- Target version changed from 0.7.2 to 0.8

- Complexity changed from Low to Medium

This is not as simple as it might sound like. Mono's String.GetHashCode() implementation uses unsafe code and the CRC32 is not as trivial as assumed. So a custom algorithm has to be written.

08/01/2010 09:04 PM - Clément Bourgeois

Mirco Bauer wrote:

> This is not as simple as it might sound like. Mono's String.GetHashCode() implementation uses unsafe code and the CRC32 is not as trivial as assumed. So a custom algorithm has to be written.

You could use something standard like SHA1 or MD5 to achieve this simply.

08/01/2010 11:53 PM - Mirco Bauer

MD5 uses 128 bits which is 4 times the size of CRC32 that uses 32 bits. Currently the nick colors uses 3 x 8 bits. Without caching this could be a bit expensive, but probably still better than a custom algorithm. Caching 2000 different nickname hashes would be 32MB of memory though, which is a lot.

08/15/2010 06:07 PM - Mirco Bauer

- Status changed from New to Closed

- % Done changed from 0 to 100

Applied in changeset commit:"94f5be71e7e39a9028e9877f967fa65e770b5d95".