Paul English
COMM 1010-003
Shirene McKay
April 29, 2013

Reflection Journal - Chapter 10 - Intercultural Relationships

1   Area Competency

Interpersonal interest

2   Goal

To improve communication with other individuals by working on social interest and connectedness in conversation and general interest.

3   Tactic

To pay attention to the relationships I have, whatever type they might be, recognize where cultural differences are to be found, and to ensure that I understand how I can improve my interpersonal interest in these relationships.

4   Journal Entries

4.1   Entry One

I have this issue where I seem to have more female friends who are completely unavailable, than those that are. I'm non-threatening, and tend to be kind to most people, or at least try to be. Apparently none of these married or committed girls thinks I will do anything to disrupt their relationships, I won't, I don't care for dishonest relationships, so they find friendship quickly. It's nice to understand people and have friends of both genders, even brief casual relationships found mostly around offices and schools. There should be no argument that genders can be considered completely different cultures. Additionally when it comes to romantic relationship status a cultural divide can be found between the long-term committed, and the non-committed. Also it can be frustrating. Still all relationships can be valuable, and can provide help and relaxation to any individual. I've earned some good friends recently both in committed and non-committed relationships, as well as amongst people I've known as acquaintances for some time.

4.2   Entry Two

So there's this thing about programmers. We're very independent, almost to the point where we all feel as though we have a solipsist mindset, completely disregarding most other people in general. We value the social interactions in the world around us, we just find that we're able to

accomplish most of our work in solo-endeavors, and we choose to communicate in the most asynchronous fashion possible to allow others to work at any schedule they find practical. This is relatively normal, and we build tools upon tools to help improve what we do. We worry about record-ability, audit-ability, and the general effectiveness of our communication tools, knowing that it is important to every situation. Email is simultaneously our most important tool for communication, and one of the most broken. This is no different for any other professional field, or personal for that matter, but as programmers we tend to emphasize it.

  With regards to email, we use it a magnitude more than the average computer user. So being capable, we also build tools around email, and around the networked machines we all have. These tools are aimed at making communication between two individuals easier, but it can never do the communicating for you.

  I have many programming friends, and we are probably the sub-most subculture involved with anything computer, or logic machine. We are paralleled only by the dedication to theory that mathematicians and philosophers have. Unlike them we remain rooted in only what can be physically achieved using what we currently have. Our culture is one that tends to avoid communication to an absurd extent, being filled to the brim with introverts, anti-social personalities, and even spectrums of high functioning autism. How on earth do we keep these machines running the world without breaking out into unparalleled civil war and division? To put it simply we don't. We can't, and those of us who try to prevent division in our field often struggle, fail, and only end up creating more division than was previously established. We embrace division, and we do our best to understand every other programmers ideas, and see if it applies to the problems we face on a day to day basis.

  Our inter-personal relationships, when working, are strictly functional. There are developers I converse with a few times a week, who I have never met, nor will I meet. I fix bugs in their software, they commit the changes. Others do the same for code that I write and share. Among those, I converse several times a day with a few non working relationship coders, who I have at some point teamed up with to create more complex and purposeful tools. These programmers share the same coding styles, and we tend to solve problems in similar ways. We often solve large problems with each other with nothing more than a mutual satisfaction in achievement, knowing that the help can always be reciprocated on days to come. Lastly there are the programmers I work with, we interact hundreds, if not thousands of times a day, where our notes, messages, and automated alerts probe each others consciousness while performing the real responsibilities we face at work. The impact of this level of communication means that I still have people sending me messages on rare occasions asking how certain code works, or reasons for it's design in jobs that we're done 2 and 3 years ago for companies I no longer work for. It's amazing that that much communication can occur, and that much impact can be made, and still most of it avoids the face to face interaction. There is nothing more valuable than the daily standup meetings we have, and the off time we spend away from machines, but to disregard the facilities we have would be a shame as well.