IST346: Debugging and Troubleshooting

Debugging Systems and Services

# The first step in debugging a problem is to…. ?????

Understand the problem

- You need to understand what the customer (or user) who reported the problem was trying to accomplish.

- In most cases the customer is expecting a specific result, but is getting an undesirable result.

- Oftentimes the data you receive is incomplete. So you must test assumptions, scope and inter-dependencies.

  Examples of "real problems"
  - I cannot print
  - The website is down
  - My computer will not send e-mail.

# Elements to Understanding a Problem
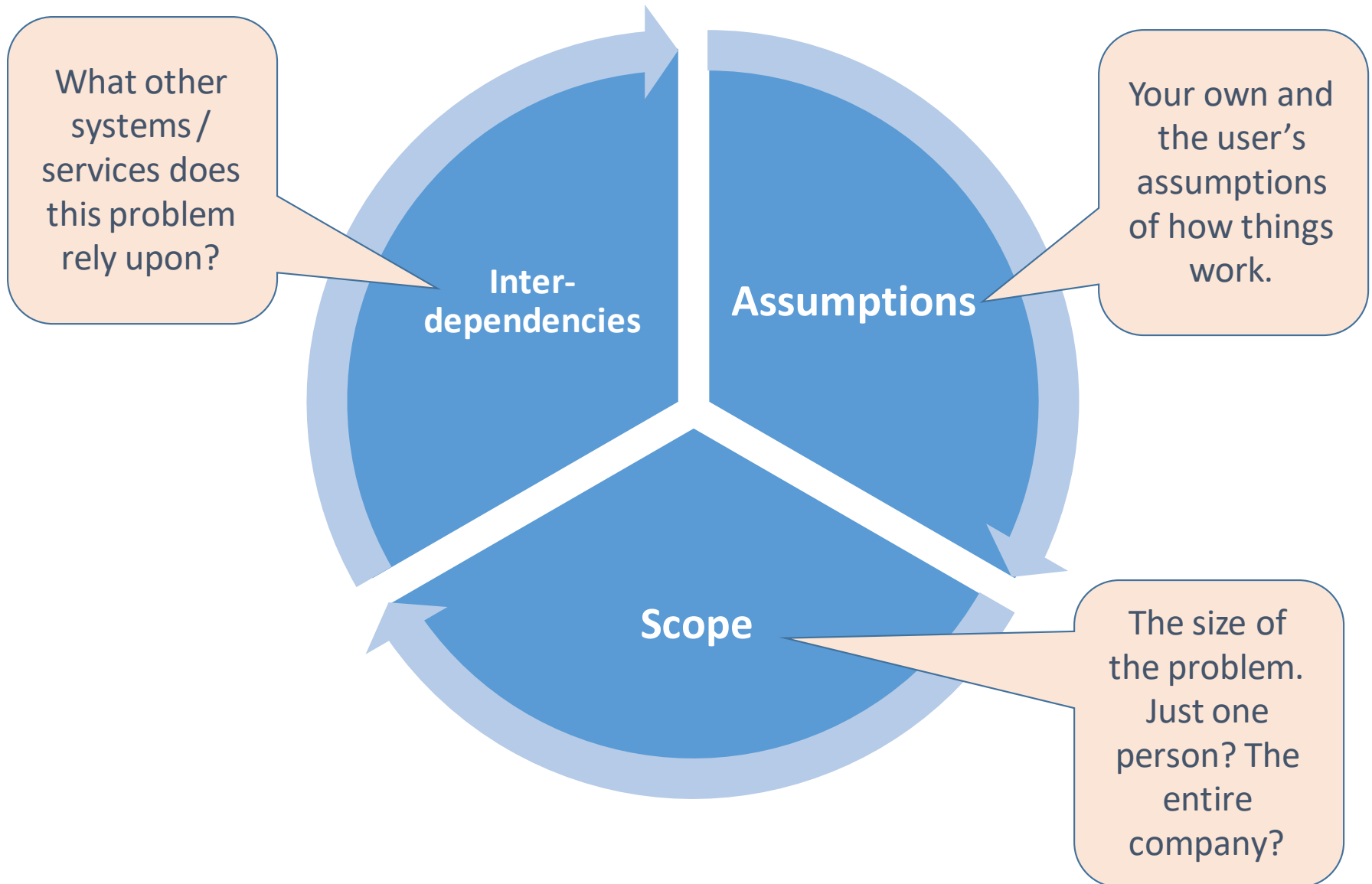
Understanding Assumptions

- Make sure you and your customer have a **mutual understanding of the situation** before attempting to debug the problem.

- Try to gather as much information from the customer as you can so that neither of you are making false assumptions.

- E.g. **I cannot print**
  - Which printer is the customer using? Don't assume a specific printer, or their default printer.
  - They might not be able to print *by design*. I.e. They don't have permissions to use a specific printer.
  - Ask the customer for specifics, of course.

Understanding Scope

- Just how **big** is the problem? What are the **boundaries**?

- Work from the **outside in**
  (widest scope, to narrowest scope)

- The scope of the problem will dictate the manner in which you react to it.

- E.g. **The Website is down**
  - Assumption: The user is in their office (not starbucks ☺)
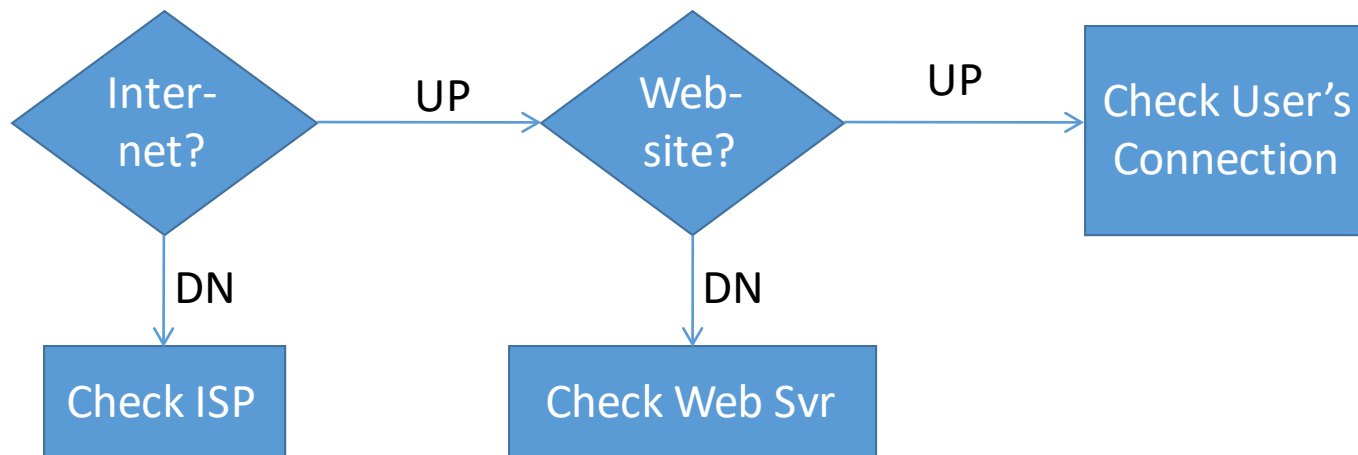
Understanding Interdependencies

- When you know the scope, next check the interdependences.

- What systems and services does that task rely upon?

- What systems and services do those rely upon?

- E.g. **"My computer will not send E-Mail"**
    - <u>Assumption, Scope</u>: Problem is organization-wide
    - E-Mail depends on SMTP to send mail.
    - SMTP needs the DNS service to work properly.
    - Check these "outside in" DNS→ SMTP → E-Mail server

Time to fix the problem

- Once you've defined the problem, and isolated it, its time to fix it.

- Quick fixes:
  - Restarting services
  - Rebooting the server dependent on services
  - Cleaning up disk space

- **These are quick-fixes and do not address the root cause!!!!**

How to find the root cause

- Be Systematic
  - Form a hypothesis or theory
  - Test and record the results
  - Make changes as necessary
- Use
  - Process of elimination
  - Successive refinement
- Review
  - Recent changes made to the system or service in question
  - Changes risk the opportunity to introduce new problems

Techniques

- **Process of Elimination**

- Remove elements of the system until the problem disappears

- Isolate and simplify the problem to identify the specific issue.

- **Successive Refinement**

- Add components to the system

- Each time verify the desired result was achieved

Fixing Things Once!

Fix it once!

- Don't use glue and duct-tape on a problem that requires wood and nails.

- When a problem seems trivial we often go for the quick fix: (The three R's)
  - Reboot computer
  - Restart services
  - Restore from backup

  - Ben Franklin Said "An ounce of prevention is worth a pound of cure."
  - Don't waste time fixing the same thing again, again, and again.

With Service Outages…

FIRST Fix it quickly!

- Get your users back up and running, ASAP
- That is the top priority first
- Do what you can to make sure it stays up until you can….

THEN Fix it *permanently*

- Spend time after the fire's out getting to the root cause and then fixing the heart of the problem
- That is the next top priority.
- Sometimes you need to dedicate a person to fixing the problem.

If you want the problem fixed for good, you need to do **both**!

Sometimes a fix is out of your control:

- Bad hardware
  - Live with it until you can replace it
  - Can you move the service elsewhere?
  - Virtualization helps here
- Some vendors write crappy software
  - Memory leaks mean that systems do not release memory when finished and thus become unstable over time.
  - Your best bet is to employ "pressure" on the vendor.
- Policy issues affect system stability
  - Examples:
    - Too many users with Root or Admin access
    - Physical access to the data center by too many people
  - Lobby to have these bad habits changed!

Example…troubleshooting in our Lab

- Problem: I cannot map a drive to [\\linuxserver\linshare](\\linuxserver\linshare) from my Windows 7 computer.
  - First, is the Linux computer powered on? (duh)
  - Can you contact the Linux server via other methods (ping, nslookup)?
    - If not, then is DNS configured properly?
  - Is the Firewall/SELinux setting turned on?
  - Is Samba installed?
    - Are the Samba services started (SMB, NMB)?
    - DNS Name setup properly in the /etc/hosts and etc/hostname files.
  - Are the share settings configured correctly (/etc/samba/smb.conf)?
  - Do you have permissions to this share?
    - Is the group configured with appropriate permissions?
    - Is your user a member of this group?
    - Does the username/password you're logged into your Win7 computer match the one setup on the CentOS5 server?

Example…troubleshooting in our Lab, pt. 2

- Problem: I cannot map a drive to [\\win2008\winshare](\\win2008\winshare) from my Windows 7 computer.
  - First, is the Windows 2008 server powered on? (duh)
  - Are you logged into your workstation with the appropriate user in your directory (Active Directory)?
  - Can you contact the Win2008 server via other methods (ping, nslookup)?
    - If not, then is the DNS server configured properly on your client?
  - Is Active Directory/DNS configured on your Win2008 server?
  - Is the share configured properly?
    - Is a folder shared with the share name of "Winshare"?
    - Are the share settings allowing you to see the share (the group)?
  - Do you have permissions to this share?
    - Did you create a Domain group to apply these permissions to?
    - Is the group configured with appropriate permissions?
    - Is your user a member of this group?