

Software Development Lifecycle (SDLC) - Support Management

April 2006 - Pragmatic Software Newsletters

Measuring the Software Development Lifecycle

Employing a solid software development lifecycle (SDLC) methodology can drastically increase your ability to deliver software projects on-time and on-budget. Once a solid SDLC methodology is in place, how do you know how efficient it is and how well it is performing? In the coming months, we will look at best practices for measuring the key indicators of SDLC, and equip you with the tools to improve your SDLC. Below are the topics to be covered in the coming months:

1. **Defect and Test Case Measurement** - Defect and Test Case Measurement is a pre-production activity that allows teams to determine the quality of their software development, and indicates when the software is ready to be released to production. [More...](#)
2. **Project Task Measurement** - Project Task measurement allows your team to determine how well individual tasks were estimated, how well they were defined, and whether items are completed on-time and on-budget. [More...](#)
3. **Overall Project Measurement** - It is important to measure overall project success by determining if the project was estimated properly, risks were identified and mitigated, requirements were correctly identified and documented, if the project was delivered on-time and on-budget. From this, we learn to provide better estimates, collect better requirements, and do better risk management. [More...](#)
4. **Support Management** - Support Ticket management is a post-production activity that allows teams to determine quality of the software release, the quality of User Guides and other documentation, and provides insight as to how well the software was architected and implemented. [More...](#)
5. **Measuring Team Goals** - For technical teams to flourish, team goals must be established and measured. Constant evaluation of the goals, and progress towards them, is critical to ensuring that team goals contribute to departmental goals.
6. **Measuring Departmental Goals** - Establishing and measuring departmental goals allow your company to grow, allow your department to identify its contribution to company growth and fosters an environment where team members thrive.

Support Management

Support Management is a post-production activity that allows teams to determine the quality of the software release, the quality of User Guides and other documentation, and provides insight as to how well the software was architected and implemented. Below are some best practices for measuring overall project success:

1. **Support Ticket Entry** - Once your software is in production, it is important to allow your clients and internal teams to submit support tickets online from your web site. How do you do this?

Create (or purchase) a support ticket system that allows your support team to answer support questions and issues. Some basic features to look for in a support ticket manager:

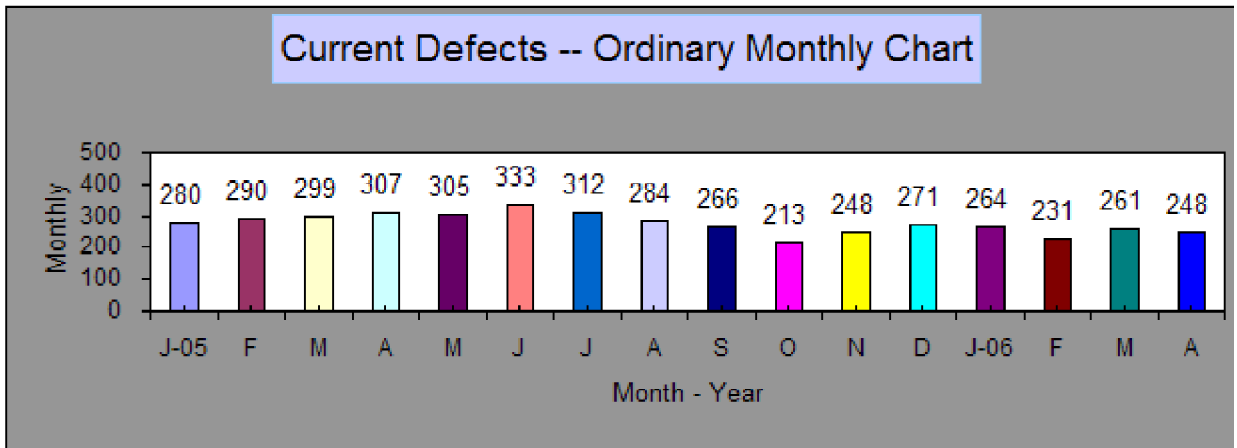
1. **Quick entry of support tickets** - You should allow your client an easy way to enter support tickets from your web site (by clicking Support from your web site or <http://www.yourURL.com/support>, etc).
2. **Email Notifications** - Once a support ticket is created, your support team should be automatically notified via email and your client should also be notified. Once the support team fixes the issue (or answers a question), the client should be notified of the resolution (via email).
3. **Checking Status** - The client should be able to check the status of their ticket online, at any time.
4. **Store Support Tickets in a Database** - All of your support tickets should be stored in a database so that you can run trending reports and do support ticket measurement.

Software Planner's [Support Ticket Manager](#) is a low cost solution for Support Ticket Entry.

2. **Support Ticket Measurement** - Once your Support Ticket Entry is in place, it is important to learn about the quality of your software by analyzing your defects and support tickets. How do you do this?

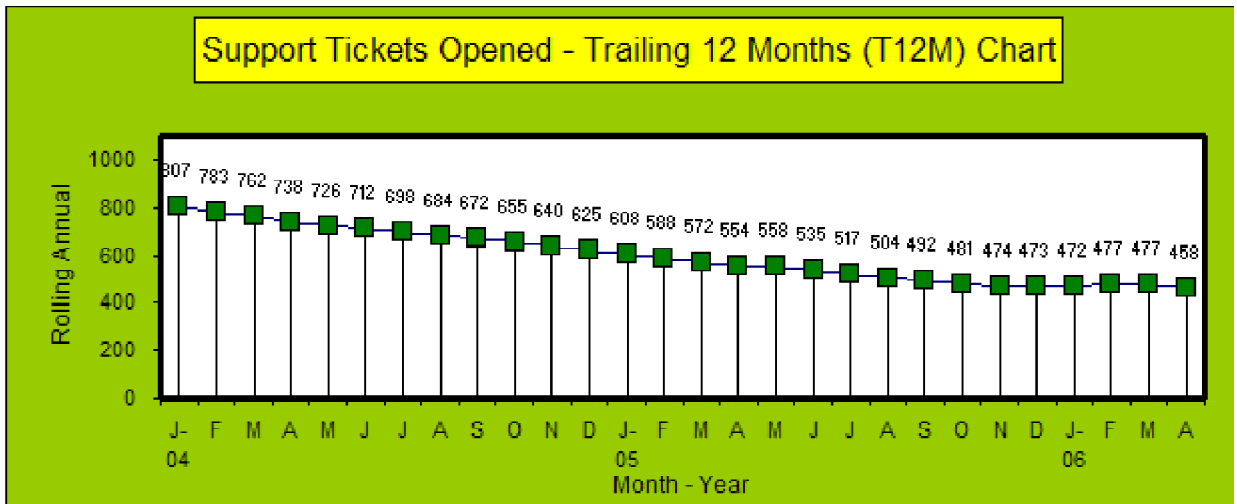
To gauge quality, it is important to track both defects and support tickets. These can be tracked in your defect tracking system, but be sure to identify which defects are support tickets vs. those that are from internal testing. If you do not have a field in your defect tracking system to track this, define a custom field that identifies the type of defect (Support Ticket vs. Internal Defect). Once this is done, begin measuring quality:

1. **Active Defect Measurement** - At the end of each day, record the number of active (open) defects. This will allow you to trend this information to see if your quality is getting better or worse. *Note: Software Planner has a [Daily Summary Report](#) that is automatically emailed to you each day so that you do not have to manually collect these statistics each day. If you are not using Software Planner, determine if this feature is available from your defect tracking solution so that you ease the collection of statistics.*
2. **Support Ticket Measurement** - At the end of each day, record the number of support tickets that came in. This will allow you to trend this information to gauge the quality of your production software release. You may find that many of the support tickets were simply questions, not actual issues. This indicates that your online help, FAQs, and/or User Guides are not robust enough to answer questions. *Note: Software Planner has a [Daily Summary Report](#) that is automatically emailed to you each day so that you do not have to manually collect these statistics each day. If you are not using Software Planner, determine if this feature is available from your defect tracking solution so that you ease the collection of statistics.*
3. **Graph your Results** - Periodically (daily or weekly), enter the statistics collected above into a MS Excel® and create graphs that allow you to spot trends. A couple of charts are helpful. First, track statistics for each month so you can see how each month trends. **Here is an example:**



Notice from the graph above, we can see that our current defects are not improving as quickly as we would like, so we may need to allocate some resources to attack the defects.

*Another helpful graph is a "trailing twelve months (T12M)" graph. This graph trends based on the past twelve months. Each point on the graph represents 12 months back, so you can use this to quickly spot if you are getting better or worse over time. **Here is an example:***



Notice from the graph above, we can see that our support tickets are decreasing over time (notice the downward trend). Here is how you read the graph:

In January 2004, there were about 807 support tickets opened per year (Feb 2003 - Jan 2004). In April 2006, there were about 458 support tickets opened in the past year (Mar 2005 - Apr 2006). This is good, points to improved production releases.

MS Excel® Template: [SP_DefectMetrics.xls](#)

3. **Supporting Your Clients** - For many software companies, their goal is to improve software quality and improve customer support. Many times, their clients will log a support ticket (or will call them directly) and will describe problem they are having. However, the support engineer may not be able to reproduce the problem. It is helpful have utilities that allow the support engineer to connect to the client's PC to determine how to reproduce the issue. How do you do this?

There are a number of programs that can allow you to connect to a client's PC. Here are a couple:

1. [Remoteus](#) - Allows the support engineer to connect to the client's PC to diagnose issues. It also allows the support engineer to reverse it, allowing the client to connect to the support engineer's PC. We like Remoteus because it is a secure connection that only lasts for the time period of the support session. Once the support session is completed, the support person can not access the client's PC.
2. [Windows MSN Messenger](#) - If the client is running Windows XP and has MSN Messenger installed, you can use the desktop sharing feature to connect to the client's PC. This free alternative is great if you are on a budget and the client does not have a problem installing MSN Messenger.

4. **Surveys** - To improve internal processes, it is important to understand how the client perceives your service. How do you do this?

One way to determine this is to perform periodic surveys. For example, if a client enters a support ticket, after the resolution is made, you can simply send the client an email asking them a few questions about their experience. Have them rate their satisfaction so that you can report this to your support team to improve the process. Keep track of surveys monthly, and make internal changes to improve customer satisfaction based on the results of those surveys.

Helpful Templates

Below are some helpful templates to aid you in developing software solutions on-time and on-budget:

- **Project Management Guidelines** - <http://www.PragmaticSW.com/Pragmatic/Templates/ProjectMgtGuidelines.rtf>
- **Functional Specifications** - <http://www.PragmaticSW.com/Pragmatic/Templates/FunctionalSpec.rtf>
- **Architectural Overview** - <http://www.PragmaticSW.com/Pragmatic/Templates/ArchitectureOverview.rtf>
- **Detailed Design** - <http://www.PragmaticSW.com/Pragmatic/Templates/DetailedDesign.rtf>
- **Strategic Planning Document** - <http://www.PragmaticSW.com/Pragmatic/Templates/StrategicPlanning.rtf>
- **Test Design** - <http://www.PragmaticSW.com/Pragmatic/Templates/TestDesign.rtf>
- **Risk Assessment** - <http://www.PragmaticSW.com/Pragmatic/Templates/Risk%20Assessment.rtf>
- **Weekly Status** - <http://www.PragmaticSW.com/Pragmatic/Templates/WeeklyStatusRpt.rtf>
- **User Acceptance Test Release Report** - <http://www.PragmaticSW.com/Pragmatic/Templates/UATRelease.rtf>
- **Post Mortem Report** - <http://www.PragmaticSW.com/Pragmatic/Templates/PostMortem.rtf>
- **All Templates** - <http://www.PragmaticSW.com/Templates.htm>
- **Prior Newsletters** - <http://www.PragmaticSW.com/Newsletters.htm>
- **Software Planner** - <http://www.SoftwarePlanner.com>
- **Defect Tracker** - <http://www.DefectTracker.com>
- **Remotus (Remote Desktop Sharing)** - <http://www.PragmaticSW.com/Remotus.asp>

About the Author

Steve Miller is the President of **Pragmatic Software** (<http://www.PragmaticSW.com>). With over 20 years of experience, Steve has extensive knowledge in project management, software architecture and test design. Steve publishes a monthly newsletter for companies that design and develop software. You can read other newsletters at <http://www.PragmaticSW.com/Newsletters.htm>. Steve's email is steve.miller@PragmaticSW.com.

Pragmatic Software Co., Inc.
383 Inverness Parkway
Suite 280
Englewood, CO 80112

Phone: 303.768.7480
Fax: 303.768.7481
Web site: <http://www.PragmaticSW.com>
E-mail: info@PragmaticSW.com