

# Building Blocks of Software Project Success

**Steven Teleki**

Director, Software Development & QA  
Multimedia Games, Inc.

Past Chair, IEEE Computer Society, Austin Chapter

# What leads to success?



St. John, Richard. Secrets of Success in 8 words, 3 minutes. TED. Feb 2005.

## What is software project **success**?

Happy, **satisfied**, profitable customers?

Happy, productive, **proud** team?

**Profitable** organization?

**Defect-free** software released **on time**?

**Success** is:

» achieving the project's objectives.

## Why aren't **more** projects successful?

Many reasons, ... plausible excuses.

Success =

- » Build the **right software** +
- » Build the software the **right way**.

This talk focuses on **Development & QA**.

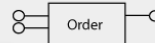
# What are the building blocks of **software project** success?



**Learn  
Always**



**Build a  
Team**



**Focus on  
Fundamentals**



**Measure &  
Improve**



**Automate**

**Building  
Blocks of Software  
Project Success**



**Plan &  
Review**



**Release  
Often**



**Manage  
Packaging**



# Build a Team

## Build a Contrarian Team and Keep it Together

### Hiring Rule #1:

» Find people with a **variety** of points of view.

### Hiring Rule #2:

» “Smart, **Disciplined**, and Gets Things Done!”

Change the assignments, but don't change the team.

**“Do not be afraid.  
Do not be satisfied with  
mediocrity!”**

Pope John Paul II.

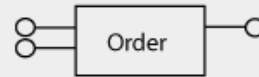
**“Mediocrity knows nothing  
higher than itself,  
but talent instantly  
recognizes genius!”**

Sir Arthur Canon Doyle

**“When two people  
always agree, one is  
unnecessary!”**

Dr. Robert H. Schuller

Pastor, Crystal Cathedral



# Focus on Fundamentals

## What are the fundamentals?

### Coupling

- » is the relationship between software parts

### Cohesion

- » the degree to which the responsibilities of a single part are logical and consistent

### Cyclomatic complexity

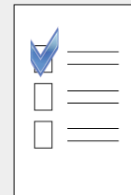
- » the number of the linearly independent paths through a program's source code

## What are the fundamentals? (continued)

**Contracts:** preconditions, postconditions, invariants, interface definitions

**Configuration** management

**Dependency** management



# Plan & Review



**“You can be sure our plan was perfect. It’s just our assumptions were wrong.”**

**Ken Olsen**

Founder & CEO  
DEC (for 35 years)  
1991

## Plan Your Work, Review, Adjust, Replan

### Why?

- » **The plan is the basis of commitments.**
- » To be successful you must be able to make commitments that you can meet—at a profit.

### What is a plan?

- » A **solution** embedded in a **list of tasks** for the work that needs to be done to achieve the desired outcome.

### How?

- » Plan in **detail**. Task length: 45-90 minutes.



# Manage Packaging

## Manage the physical design and packaging

### Physical design

- » addresses the issues surrounding the physical entities of a system (e.g., files, directories, and libraries) as well as organizational issues.



# Release Often

## Release early! Release often!

### Release

- » The action of making a product available for general purchase.
- » ...from one phase to the next...
- » Make it mean something!
- » Make it mean: **DONE!**



# Automate

## Automate, Automate, Automate!

There is no reason for a human to do something twice that can be automated!

### Automate:

- » Build (multiple times a day)
- » BVT, FVT, SVT (daily)
- » End-to-End user test scenarios (daily)
- » Release processes (weekly)



# Measure & Improve

## Track Effective On-Task Time (EOT)

The time **effectively** spent on project work.

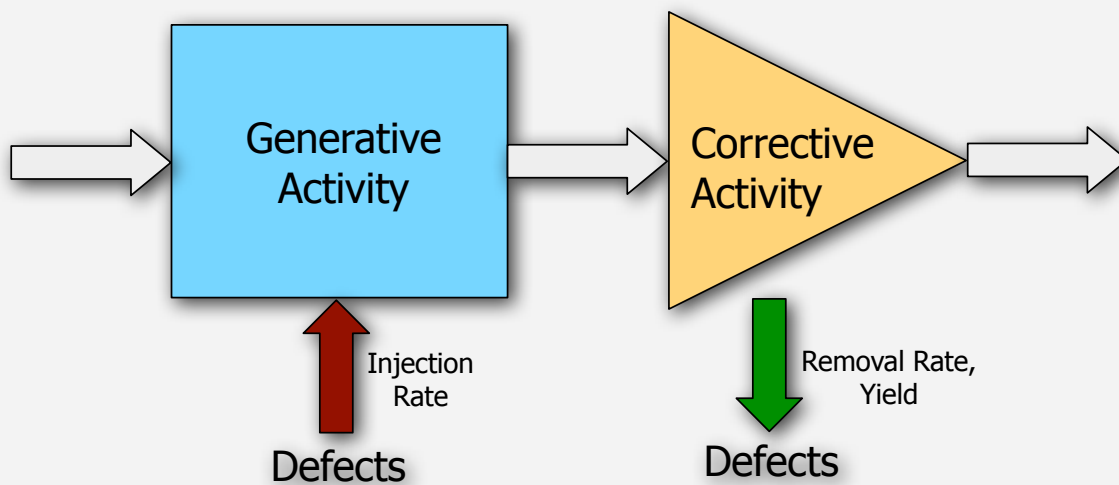
Doesn't include:

- » Reading email (usually even if it is project related)
- » Meetings (except well-defined project meetings)
- » Lunch time, breaks, phone conversations, etc.

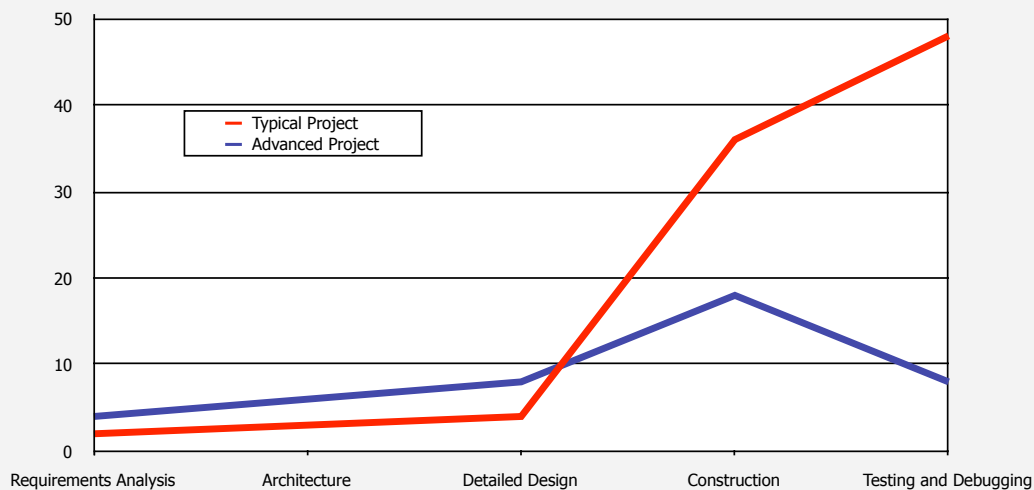
Measure your EOT per week.

- » Best organizations in the world get **20+ hrs/week**.
- » You may only get about 3-5 hrs/wk the first week.  
You should get up to 15 hrs/wk in a few weeks.

## Assemble Processes with the Process Building Block



## Differentiate Work Phases



Advanced software development approaches require more work during the early stages of the project to eliminate enormous amount of unnecessary work in the later stages of a project.



# Learn Always

## DISTINCT ... OR EXTINCT!

**“If there is nothing very special about your work, no matter how hard you apply yourself, you won’t get noticed and that increasingly means you won’t get paid much, either.”**

Michael Goldhaber, Wired

## Focus on Your Talents

### What is a talent?

*“A talent is a recurring pattern of thought, feeling, or behavior that can be productively applied.”*

Every role performed *at excellence* requires talent.

*“Michelangelos of housekeeping.”*

Buckingham, Marcus, Curt Coffman. First, Break All The Rules. Simon & Schuster. NY, NY. 1999.  
Peters, Thomas J. The Circle of Innovation. Random House. New York, NY. 1997.

**“In every human activity  
the law of the farm  
governs.”**

**“There is no cramming  
on the farm.”**



Corn Granary, Eastern Europe, 2007

Stephen R. Covey



# Conclusion



**“If things seem under control, you are just not going fast enough!”**

Mario Andretti

race car driver

**Your Letters & Comments are Welcome!**

Steven Teleki:

» [steve@teleki.net](mailto:steve@teleki.net)

Visit: <http://steven.teleki.net/>

» Software Development Reading List

» Slides from this and previous talks