

allurent



Automation of Flex Building and Unit Testing or Continuous Integration with Flex, FlexUnit, and Ant

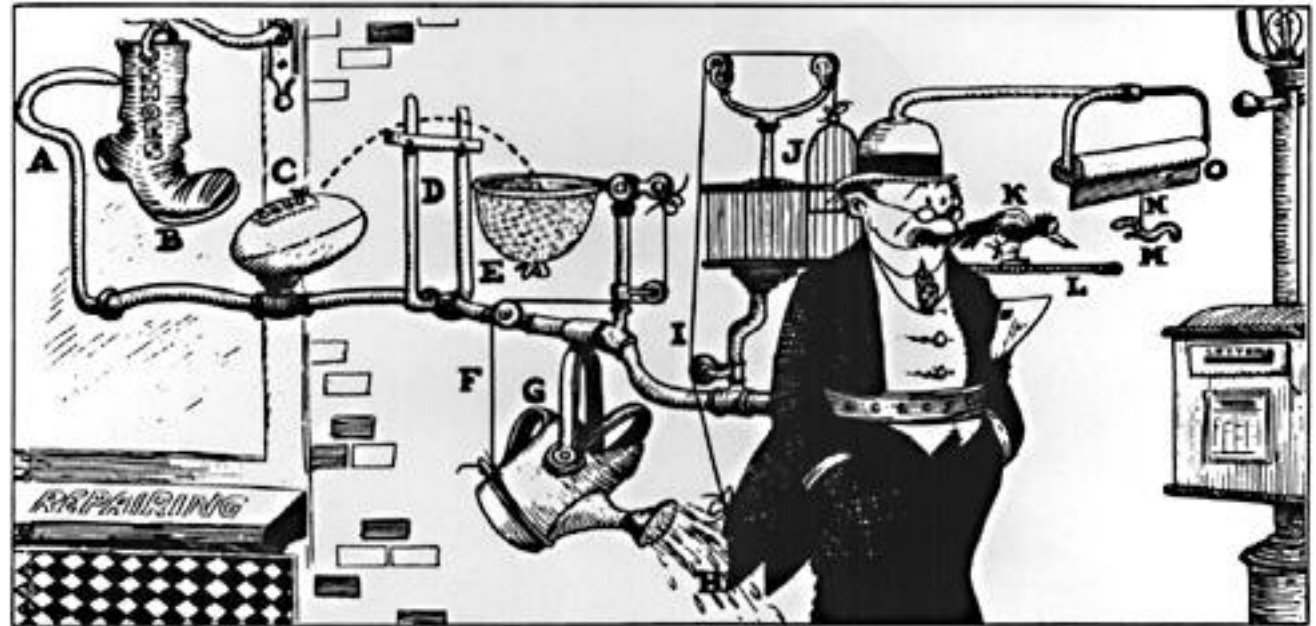
Daniel Rinehart

Software Architect

Allurent, Inc.

Is your build process like a Rube Goldberg contraption?

- Manual
- Fragile
- Missing



Keep You From Forgetting To Mail Your Wife's Letter RUBE GOLDBERG (tm) RGI 049

Goal of this talk is to introduce tools and techniques to create a continuous integration based build process.

- Centralized
- Automated
- Self-Testing



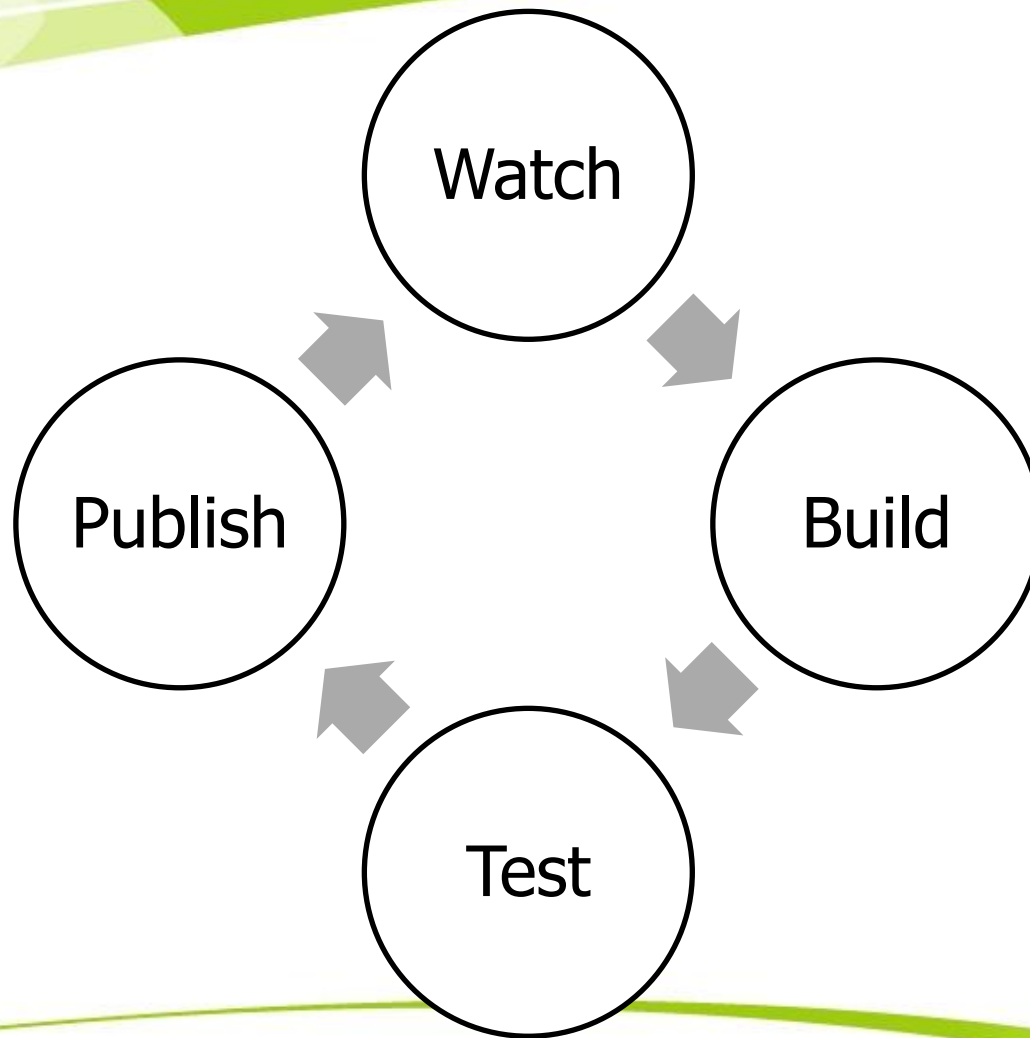


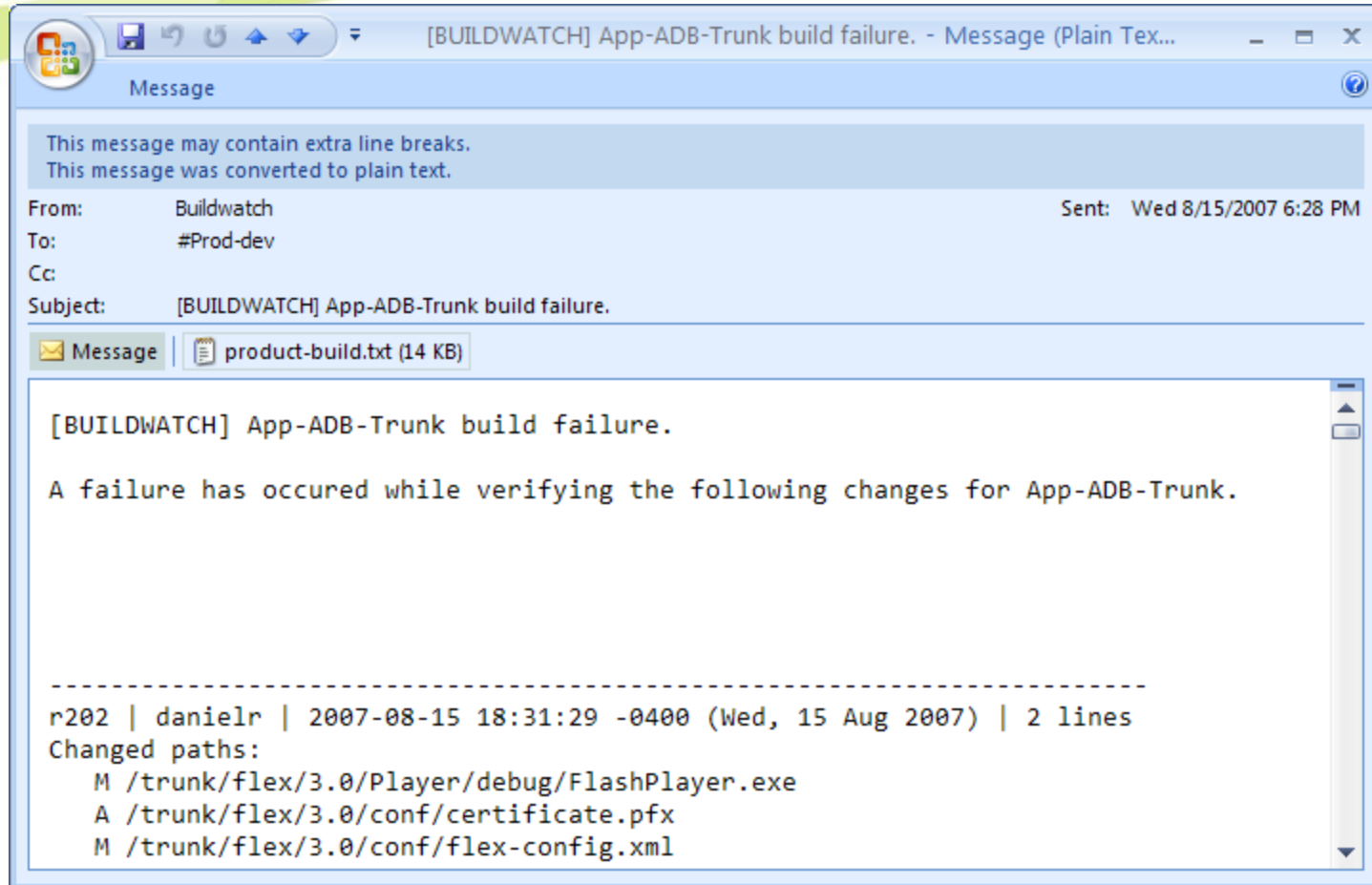
Continuous Integration is a software development practice where members of a team **integrate** their work frequently.

Each integration is verified by an **automated** build and test that detects integration errors as quickly as possible.



Continuous Integration Cycle





```
product-build.txt - Notepad
File Edit Format View Help
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/app/ui-adb/flex2/com/allur
ent/arcx/adb/controller/AdbController.as(131): col: 62 Error: Call to a possibly
undefined method systemTrayHandler.
    [java]
    [java]         public var systemTrayHandler:SystemTrayHandler = new
SystemTrayHandler();
    [java]
    [java]
    [java]
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/app/ui-adb/flex2/com/allur
ent/arcx/adb/controller/AdbController.as(131): col: 62 Error: Call to a possibly
undefined method systemTrayHandler.
    [java]
    [java]         public var systemTrayHandler:SystemTrayHandler = new
SystemTrayHandler();
    [java]
    [java]
    [java]
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/app/ui-adb/flex2/com/allur
ent/arcx/adb/controller/AdbController.as(46): col: 19 Error: Definition
system:SystemTrayHandler could not be found.
    [java]
    [java]         import system.SystemTrayHandler;
    [java]
    [java]
    [java]
BUILD FAILED
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/app/build.xml:42: The
following error occurred while executing this line:
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/app/build.xml:20: The
following error occurred while executing this line:
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/dev/tools/build/build2-com
mon-targets.xml:67: The following error occurred while executing this line:
/allurent/relog/buildwatch/trunk/app/adb/builds/9490/checkout/dev/tools/build/build2-com
mon-tasks.xml:71: Java returned: 4

Total time: 1 minute 30 seconds
```

Allurent Buildwatch

Buildwatch is a continuous build system used to help quickly detect product regressions. Every 5 minutes it checks for code changes and, if a change exists, it re-builds the product and runs all junit and flexunit tests.




Alert email is sent if a failure is encountered in any steps.

For more information see the [Buildwatch wiki page](#)

Current Status

Description	Status	Last Build Time	Platform
App-ADB-Trunk	PASS	09/16 05:27 PM	Linux 2.6.9-42.ELsmp
App-Client-Trunk	FAIL	09/17 12:19 PM	Linux 2.4.21-32.0.1.ELsmp
App-Trunk-PlatformDist	PASS	09/14 03:46 PM	Linux 2.4.21-32.0.1.ELsmp
App-Test-Trunk	PASS	09/16 07:31 PM	Linux 2.6.11-1.14_FC3
ArcSuite-Trunk	PASS	09/14 10:00 AM	Linux 2.4.21-32.0.1.ELsmp

Key

-  Product builds successfully and there are no test failures.
-  Product builds successfully, but there are test failures.
-  Product does not build successfully.

Allurent Release Engineering

* *Active nightly builds.*

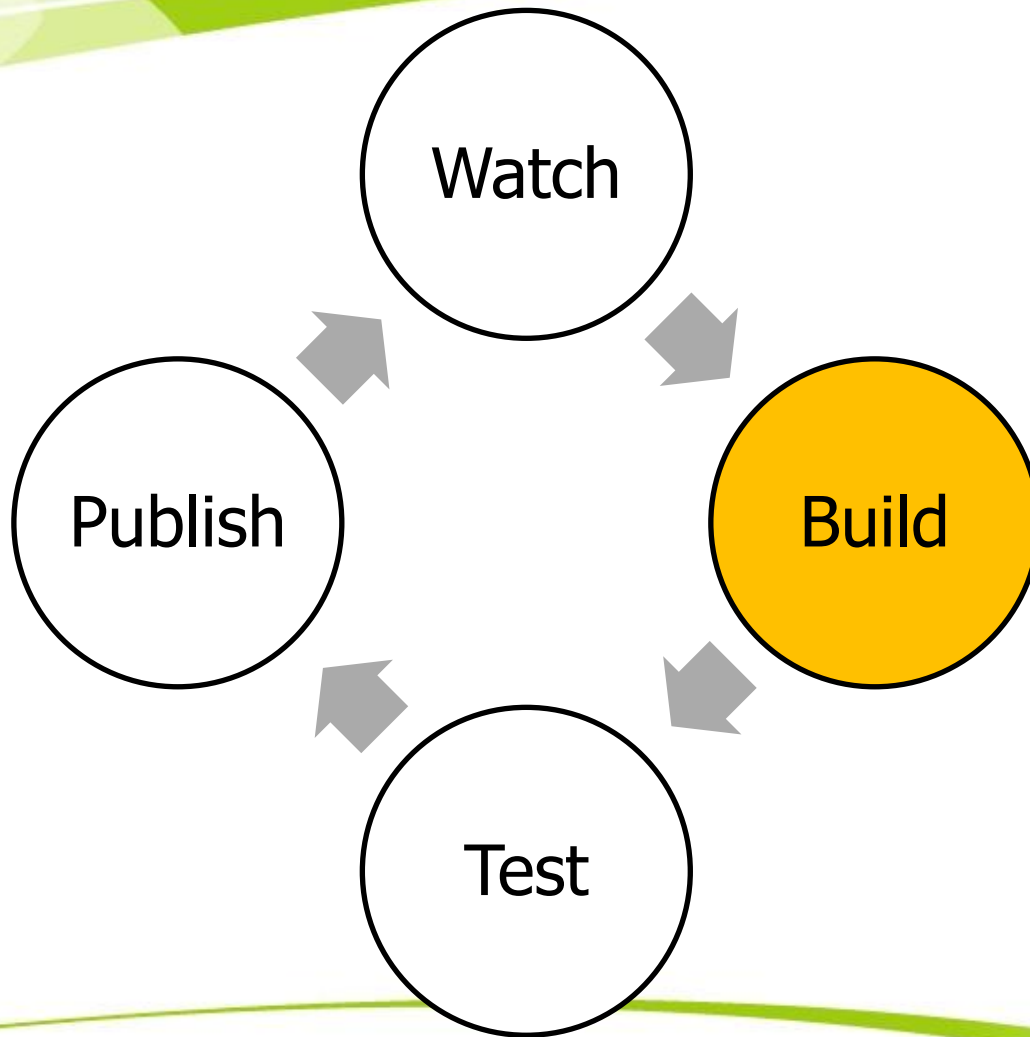
PRODUCTS - Flex 3.0

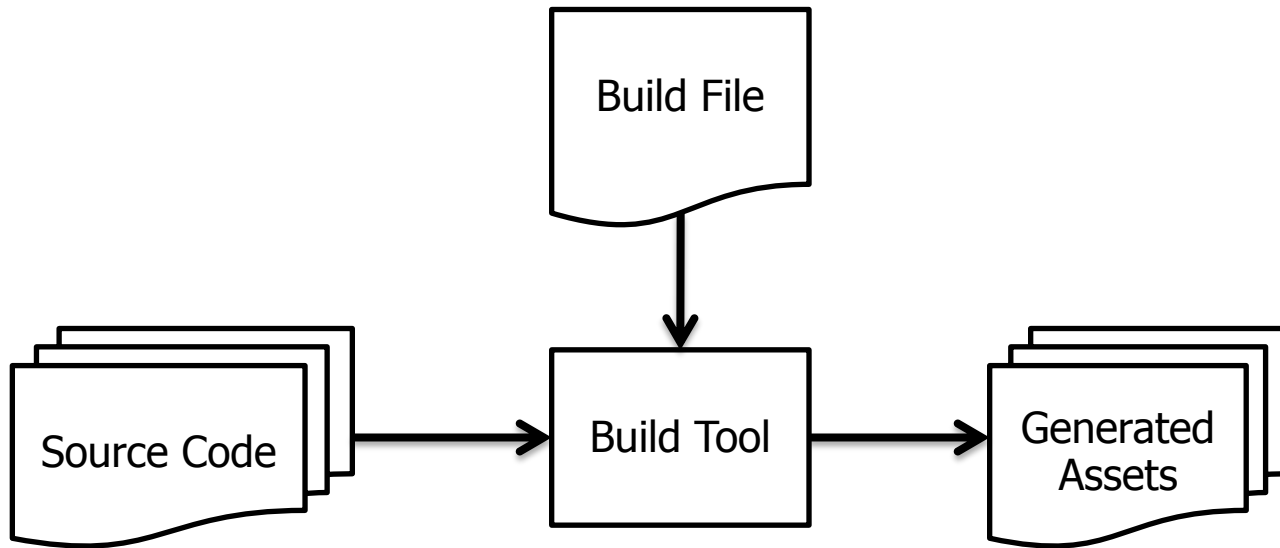
ADB			
	trunk *	Dists	<< Download Dist

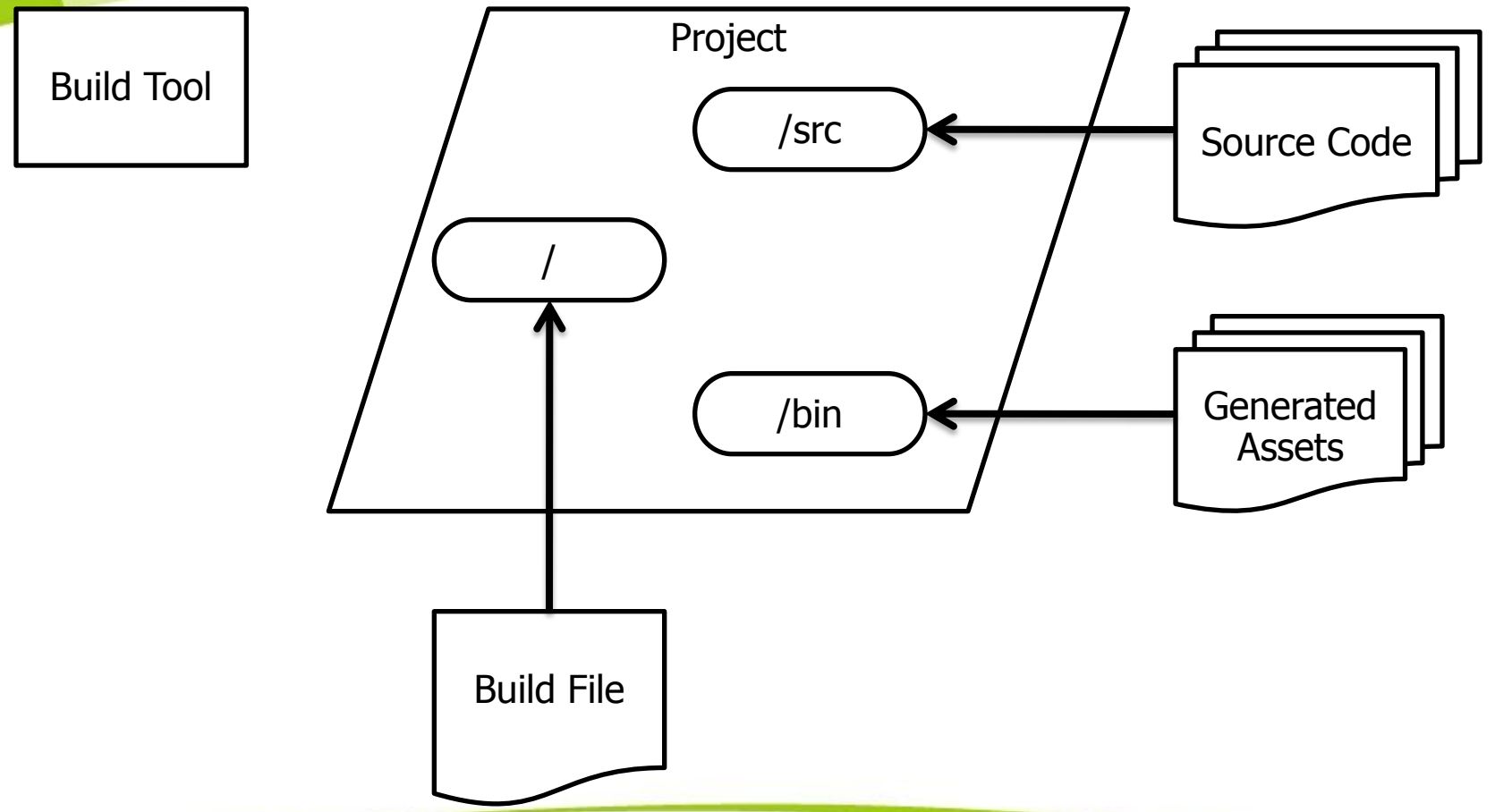
PRODUCTS - Flex 2.0

Demo Installer				
			Installation Instructions	
ArcPlatform				
	trunk *	Dists	Docs	
	2.0.x	Dist	Docs	
XPM-Common				
	trunk *	Dists		
	2.0.x	Dist		
XPM-Checkout				
	trunk *	Dists	Docs	Test Server
	2.0.x	Dist	Docs	Test Server
XPM-Display				
	trunk *	Dists		Test Server
Allurent Sales Demo				
	2007 - trunk *	Dists		Test Server

Continuous Integration Cycle









Build Automation Tool

- Widely Used
- Extensive Documentation
- Cross Platform
- Robust
- XML Based



Project

- Root element of a build file
- Defines steps of build process
- Ideally one-to-one correspondence to real projects

Target

- Logical chunk of work
- Can depend on other targets
- Works like a script

Property

- Holds a constant textual value or file location
- Accessed with `${}` syntax
- Environment or custom defined

Task

- Individual command
- Accepts arguments to control behavior
- Used in targets



Ant Hello World Demonstration

```
<project name="HelloWorld" default="build">  
  <property name="message" value="Hello World!"/>  
  <target name="init" description="Initialize the build">  
    <tstamp/>  
  </target>  
  <target name="build" depends="init"  
    description="Build the Hello World project">  
    <echo message="${message} Today is ${TODAY}"/>  
  </target>  
</project>
```


Executed build file	Buildfile: build.xml
Name of target being run	init:
Name of target being run	build:
Task name and output	[echo] Hello World! Today is September 17 2007
Result of the build	BUILD SUCCESSFUL
Diagnostic information	Total time: 0 seconds

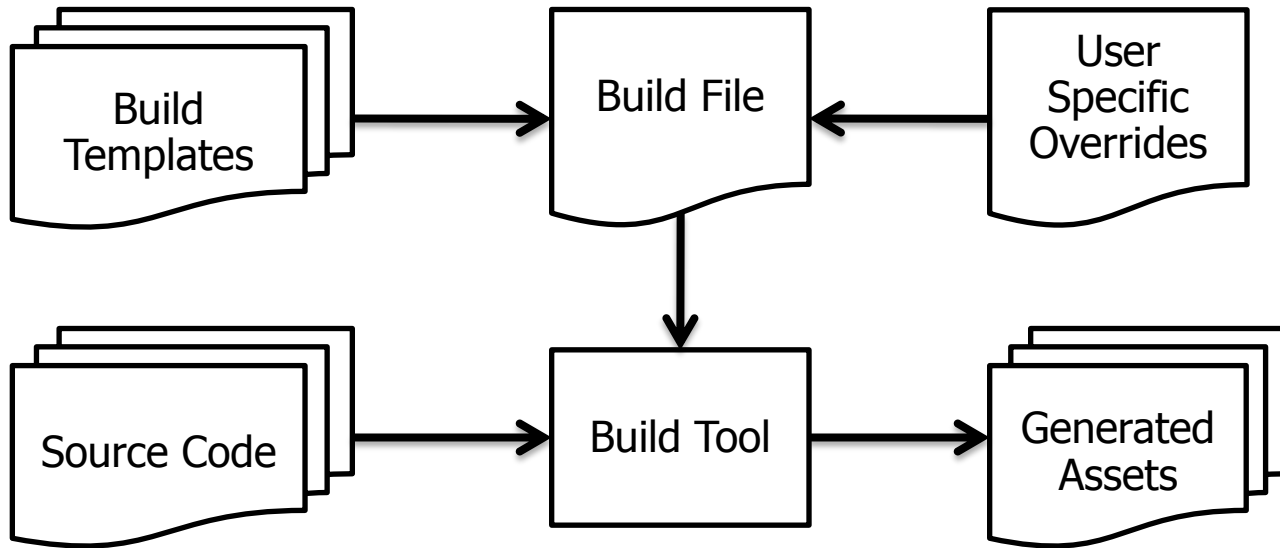


Ant mxmhc Demonstration

```
<!-- Using the Flex Ant Tasks provided by Adobe Labs -->
<project name="SampleApplication" default="build">
  <taskdef resource="flexTasks.tasks"
    classpath="${basedir}/flexTasks/lib/flexTasks.jar"/>
  <property name="FLEX_HOME" value="C:/Talk/Flex2.0.1HF2"/>
  <target name="build">
    <mxmlc file="src/${ant.project.name}.mxml"
      output="bin/${ant.project.name}.swf">
      <load-config
        filename="${FLEX_HOME}/frameworks/flex-config.xml"/>
      <source-path path-element="${FLEX_HOME}/frameworks"/>
    </mxmlc>
  </target>
</project>
```



- Keep source and output separated
- Don't do work you've already done
- Provide reasonable defaults
- Centralize behavior
- Predictability





Build Templates

- Encapsulates Best Practices
- Handles Flex Applications, Libraries, and Documentation
- Cross Project Dependencies
- Multiple Project Builds

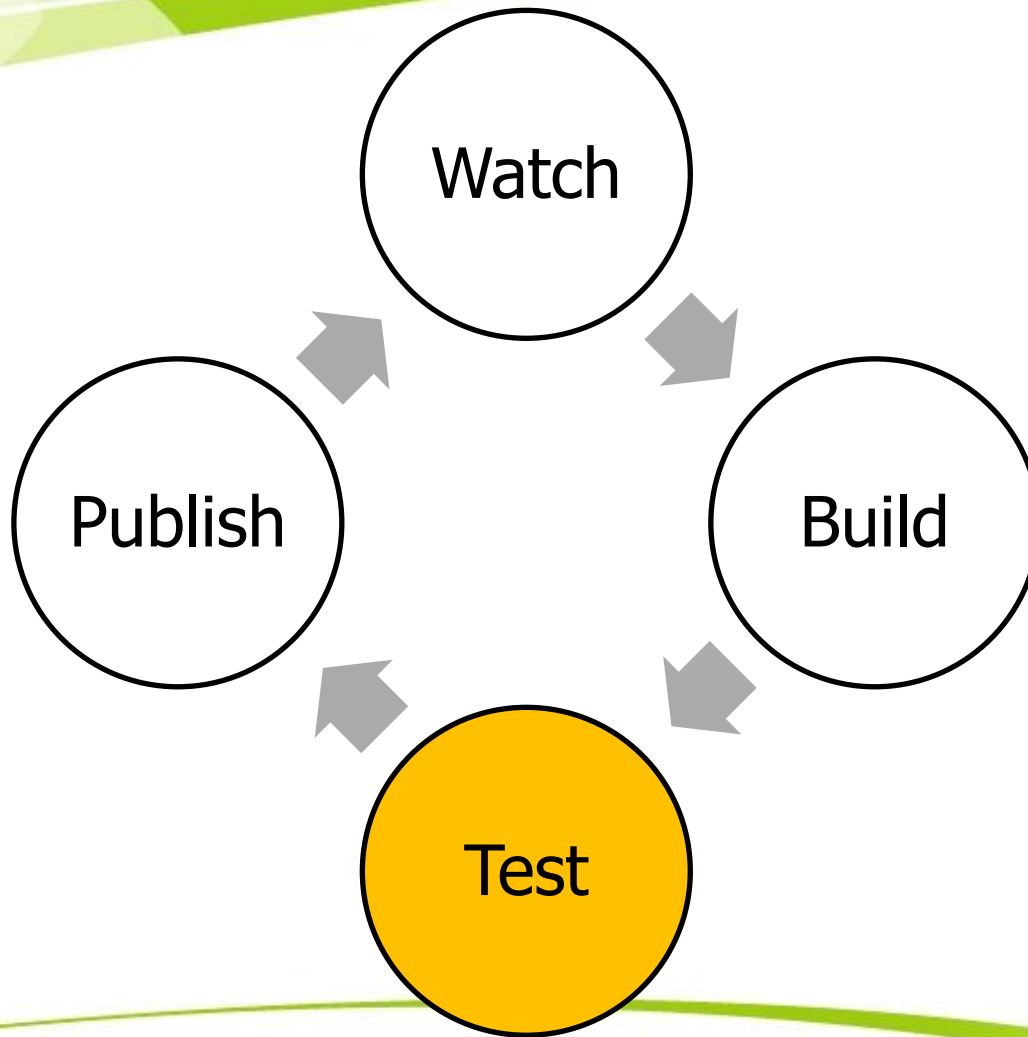


Antennae Demonstration

```
<project name="SampleApplication" default="build">  
  <property name="root.dir" location="../"/>  
  <property file="${root.dir}/build-user.properties"/>  
  <import file="${root.dir}/tools/build-imports.xml"/>  
  <path id="flex2.lib.path"/>  
  <target name="build" depends="init,flex2-application"  
    description="Build the application" />  
</project>
```




Continuous Integration Cycle



Build Verification

- Supports Flex and ActionScript
- Asynchronous Testing Facilities
- Graphical Test Results

Adobe Flash Player 9

File View Control Help

Test Runner

Running... Run: 2/2 Errors: 0 Failures: 1

Failures **All Tests**

testThatFails - SampleTest

Stack Trace

```
Error: expected:<true> but was:<false>
  at flexunit.framework::Assert$/::failWithUserMessage()
  at flexunit.framework::Assert$/::failNotEquals()
  at flexunit.framework::Assert$/assertEquals()
  at SampleTest/testThatFails()
  at flexunit.framework::TestCase/::runTestOrAsync()
  at flexunit.framework::TestCase/runMiddle()
  at flexunit.framework::ProtectedMiddleTestCase/protect()
  at flexunit.framework::TestResult/::doProtected()
  at flexunit.framework::TestResult/::doContinue()
  at flexunit.framework::TestResult/::doRun()
  at flexunit.framework::TestResult/run()
  at flexunit.framework::TestCase/runWithResult()
  at flexunit.framework::TestSuite/::runTest()
  at flexunit.framework::TestSuite/runNext()
  at flexunit.framework::TestSuiteTestListener/handleTimer()
  at flash.utils::Timer/flash.utils:Timer::_timerDispatch()
  at flash.utils::Timer/flash.utils:Timer::tick()
```



FlexUnit Demonstration



FlexUnit Automation

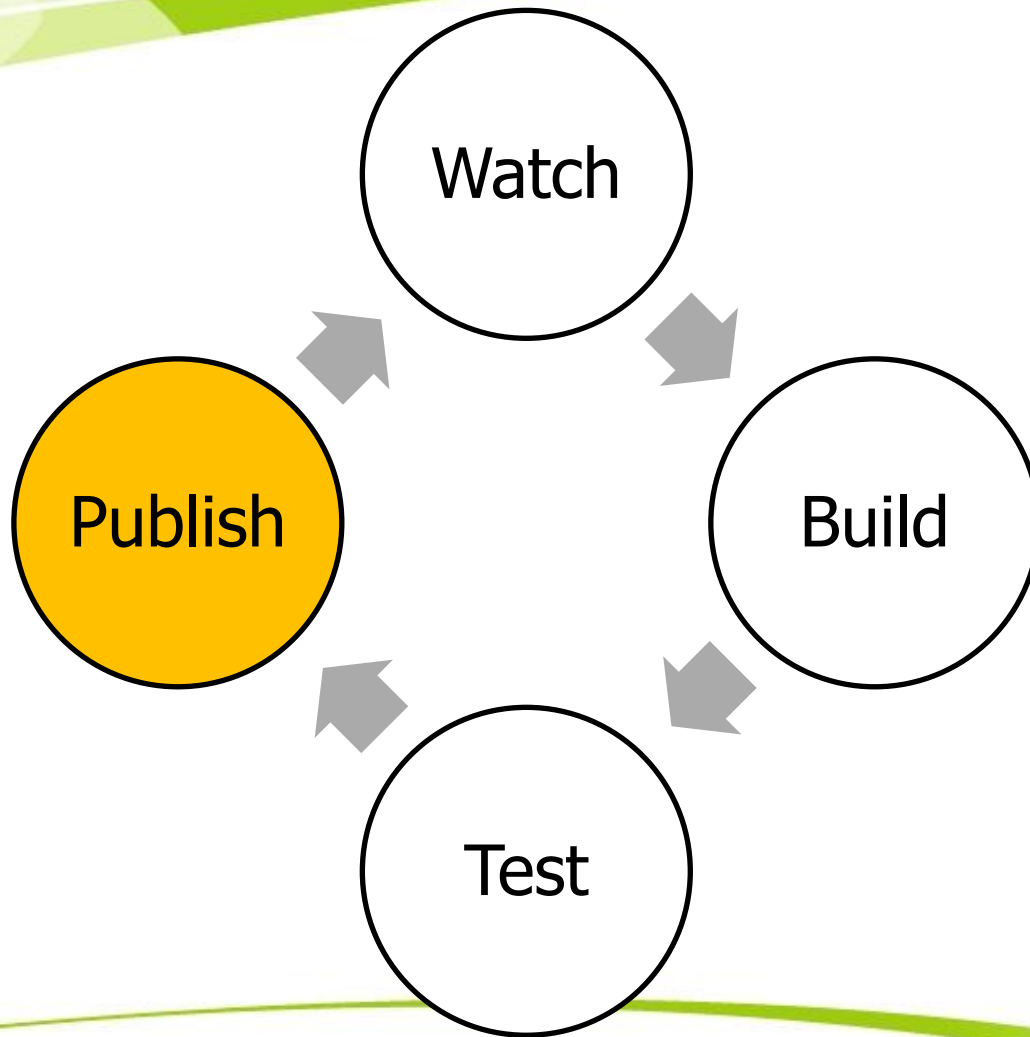
- Test Suite Generation
- Test Filtering
- Actionscript to Java Result Reporting



Antennae Demonstration

```
<project name="SampleTestApplication" default="test">
  <!-- Shared imports -->
  <import file="${root.dir}/lib/build-assets.xml" />
  <path id="flex2.lib.path">
    <pathelement location="${flexunit2.swc}"/>
    <pathelement location="${arc-flexunit2.swc}"/>
    <!-- Include asset under test here -->
  </path>
  <target name="build" depends="init,flex2-test-application"
    description="Build the test application"/>
  <target name="test" depends="build,test-flexunit2"
    description="Run the test application"/>
</project>
```

Continuous Integration Cycle

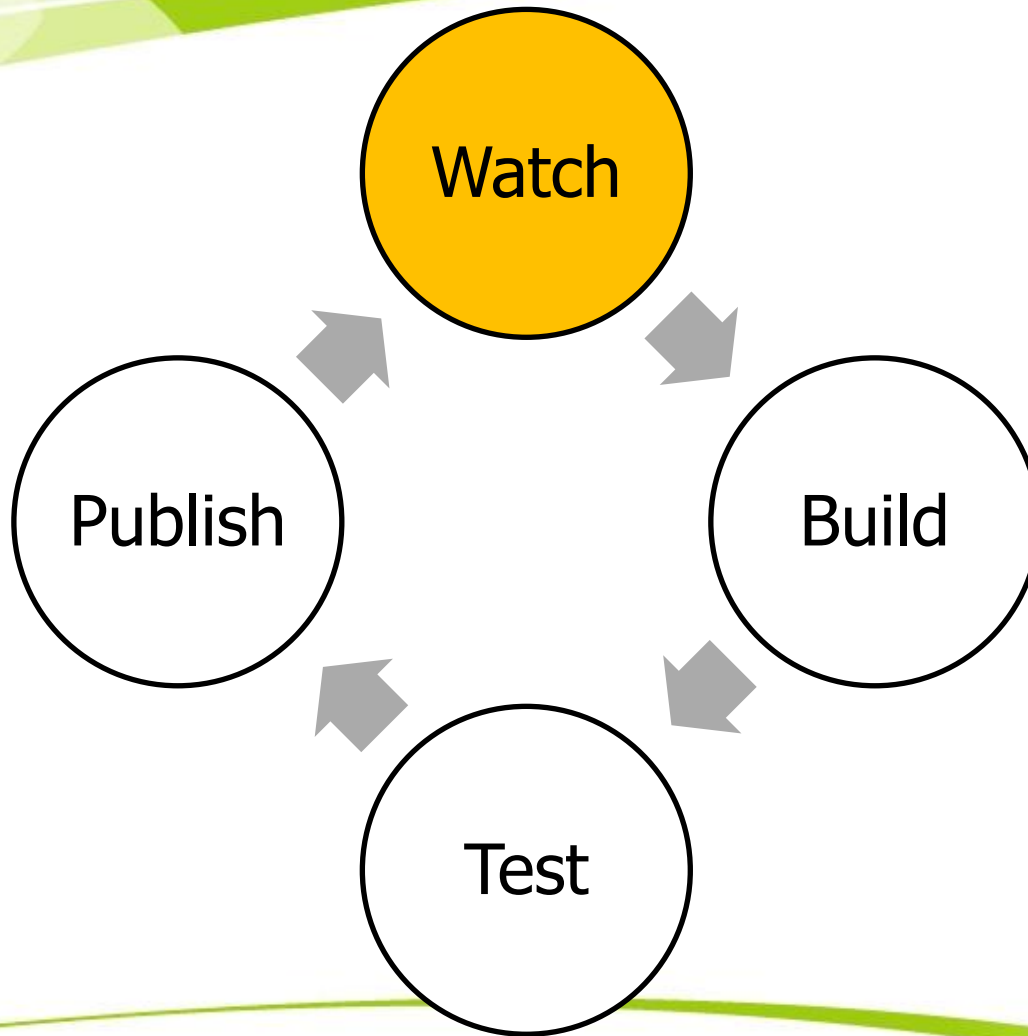




Ant Demonstration



Continuous Integration Cycle





Key Concepts

- Source code repository
- Clean build machine
- Runs after every commit

Popular platforms

- CruiseControl
- Apache Continuum
- Many others



- Ant
 - <http://ant.apache.org/>
- FlexUnit
 - <http://code.google.com/p/as3flexunitlib/>
- Antennae
 - <http://code.google.com/p/antennae/>
- Continuous Integration Matrix
 - <http://docs.codehaus.org/x/gCk>
- Flex Ant Tasks
 - http://labs.adobe.com/wiki/index.php/Flex_Ant_Tasks



Q & A



danielr@neophi.com

<http://danielr.neophi.com/>