software AG

Boosting Agile Methodology with webMethods BPMS

Sami Morcos
Chief Solution Architect, Software AG Enterprise Architect Office

Rupinder Singh
Director, Software AG Global Consulting Services

PROCESSWORLD.2011

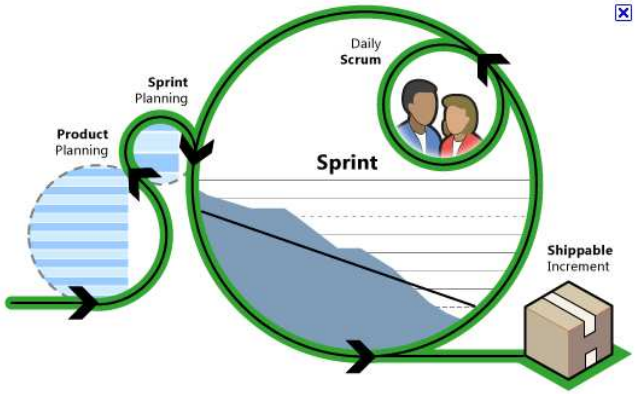
Topics

- Agile Overview
- Why Agile
- An Agile Process: Scrum in a Nutshell
- Boosting Agile with High Power BPMS Environment
- Customers using an Agile methodology with webMethods
- Agile Development: A Consultant's Perspective

Agile Overview: Definition

A group of software development methodologies based on iterative, incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional team

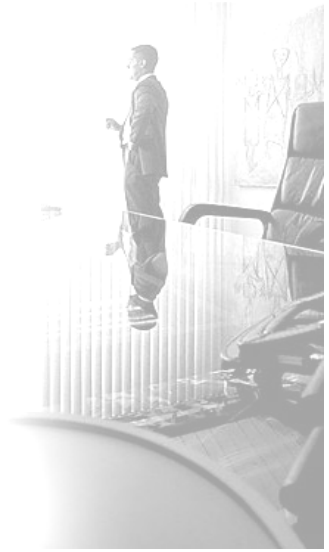
A process that focuses on delivering highest business value in shortest time.



Agile Overview: History

- Rapid Application Development (RAD) - 1990
- Scrum (1995)
- Extreme Programming - 1996
- Adaptive Software Development (ASD) - 1999
- Crystal Clear - 2004
- Pragmatic Programming - 1999
- Feature Driven Development (FDD) - 2002
- Dynamic Systems Development Method (DSDM)

- Agile Manifesto. February 2001



Agile Overview: Seven Principles

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (wks rather than months)
4. Working software is principal measure of progress
5. Sustainable, Adaptive Software Development
6. Face-to-face conversation is best form of communication (co-location)
7. Self-organizing teams: Projects are built around motivated individuals, who should be trusted

Why Agile?

- Better Customer Satisfaction
- Stronger alignment between business and IT
- Faster Development
- Competitive Advantage
- Better Quality Solution
- Faster Time to Develop, Market
- More Sustainable

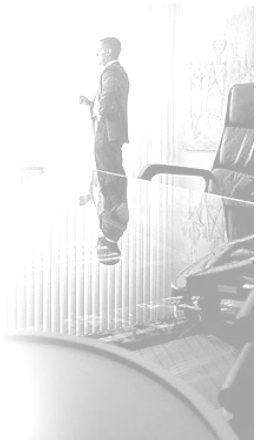
Who has Used Agile

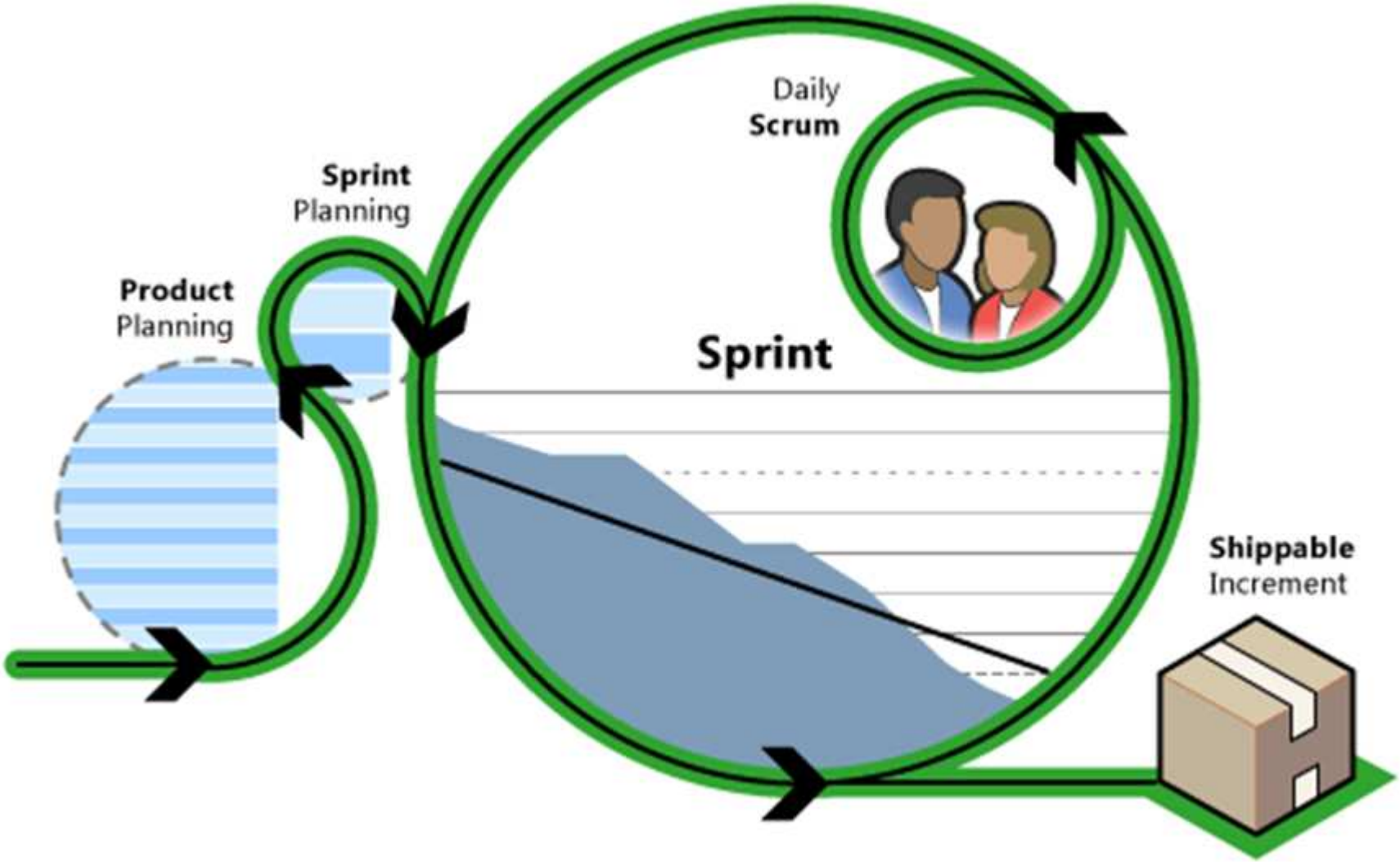
- Software AG
- Microsoft
- Yahoo
- Google
- Lockheed Martin
- Siemens
- Sun
- Nokia
- BBC
- Electronic Arts
- High Moon Studios
- Philips
- Capital One
- Intuit
- IBM
- Oracle

- Amtrak
- Salesforce.com
- Adobe
- Toyota
- SAP
- GE Medical
- Intuit
- Turner Broadcasting
- Nielsen Media
- First American Real Estate
- BMC Software
- Amtrak
- John Deere
- Lexis Nexis
- Sabre
- Time Warner

Scrum in a Nutshell

- An Agile methodology / process
- Coauthored by Jeff Sutherland, Ken Schwaber
- Roles: Self-organizing teams – total 5 to 9
  - Product Owner, Scrum Master, The Team
- Requirements: Captured in a list called Product Backlog
  - Determined, prioritized by Product Owner
- Development Iteration: Sprint
  - Sprint Planning > Sprint Backlog
  - Product is designed, developed, & tested during a Sprint
  - Typical duration: 2 – 4 weeks
  - Output: usable software
- Key Ceremonies
  - Sprint Planning, Sprint Review, Sprint Retrospective, Daily Scrum
- Key Artifacts
  - Product backlog, Sprint Backlog, Sprint Introspective, Burn-down Chart




Daily
Scrum
Sprint
Planning
Product
Planning
Sprint
Shippable
Increment

Can Technology Impact Agile Development

- The three critical pillars of software development
  - People
  - Methods
  - Technology
- The Agile position on technology
- High-power technology can improve quality and speed of agile

High Power Enabling Technology: What to look for

- Supports Rapid Development Environment
  - To helps shorten Iteration / Sprint duration
  - Shorter time to build and test user stories
- Flexible Configuration-based Development
  - Visual development, declarative UIs, wizards, less coding
  - Allow coding, when needed, in any language
- Foster User Engagement, by enabling
  - Rapid creation, testing of UIs on different mediums – browsers, Mobile Devices, etc
  - Rapid creating, testing of Human Interactions, workflow development
- Strong support for Reuse
  - Of pre-built capabilities
  - External services, built by other apps capabilities
  - Newly configured services

High Power Enabling Technology: What to look for (continued)

- Strong Integration to Other Applications, by supporting
  - Interoperability standards
  - Productivity tools
- Strong Integration to External Partners, by supporting
  - Interoperability standards
  - Productivity tools
- Strong Testing Capabilities
- Strong Team Development Capabilities

Boosting Agile with webMethods Technology

- **ESB: Automate, Expose, & Re-Use Services and Interfaces**
  - Rapid connectivity, configuration, and testing of interfaces
  - Productivity gains by re-using pre-built services
  - “working solution” demands enterprise integration quality

- **Process Intelligence:** Automate Visibility Requirements
  - Rapid, visual UI design, development – Including mobile devices
  - Rapid instrumentation of source data

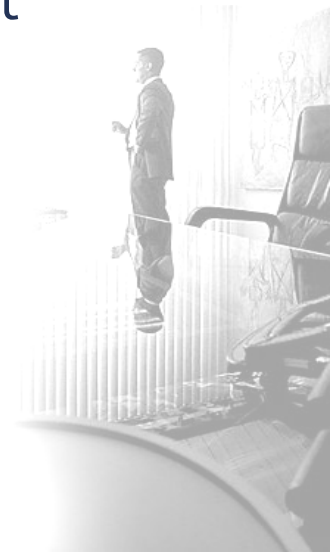
- **BPM Orchestration, Human Workflow: Automate Processes**
  - Rapid, wizard-driven human workflow rules
  - Rapid configuration of UI artifacts
  - Code-less Composite Application Development
  - Code-less BPM tracking, monitoring, & orchestration

Customers Using Agile Methodology w/Software AG webMethods

- BC Government: Exemption Request Process
- Amtrak
- Lockhead Martin
- MITRE
- Northrop Grumman
- SAIC
- Department of Defense
- Large Manufacturer of Medical Devices
- Large Utility Company
- Large Bank (payment process)
- Railway Company
- Large Insurance

Agile Suitability: Factors Negatively Impacting Agile

- Forcing Agile on Development teams
  - buy-in, trust, exec support
- Distributed development efforts (non co-located teams)
  - Distributed Agile - Scrum of Scrums
- Mission Critical apps: failure is not an option at any cost
  - E.g. process control in nuclear plant
- Applications with numerous interfaces
- Large-scale development efforts



Agile Development: A Consultant's Perspective

Rupinder Singh,
Director, Software AG Global Consulting Services

Agile Development: Important Facets

- Untethered development
  - Developers more agile
  - Multiple iterations and releases possible
  - Relative independence promotes agile culture
- Test Driven
  - Unit Testing
  - Regression Testing
  - Continuous Integration
- Automation
  - Less reliance on human factors
  - Quicker turnaround
  - Better integration in large initiatives with disparate technologies

Agile Development: Important Facets

- Software Version Control
  - Tighter integration into VCS systems
  - Visibility into changes and impact
- Distributed Deployment
  - Automated
  - Repetitive deployment based on versions
  - Smoke Tests
  - Continuous Integration

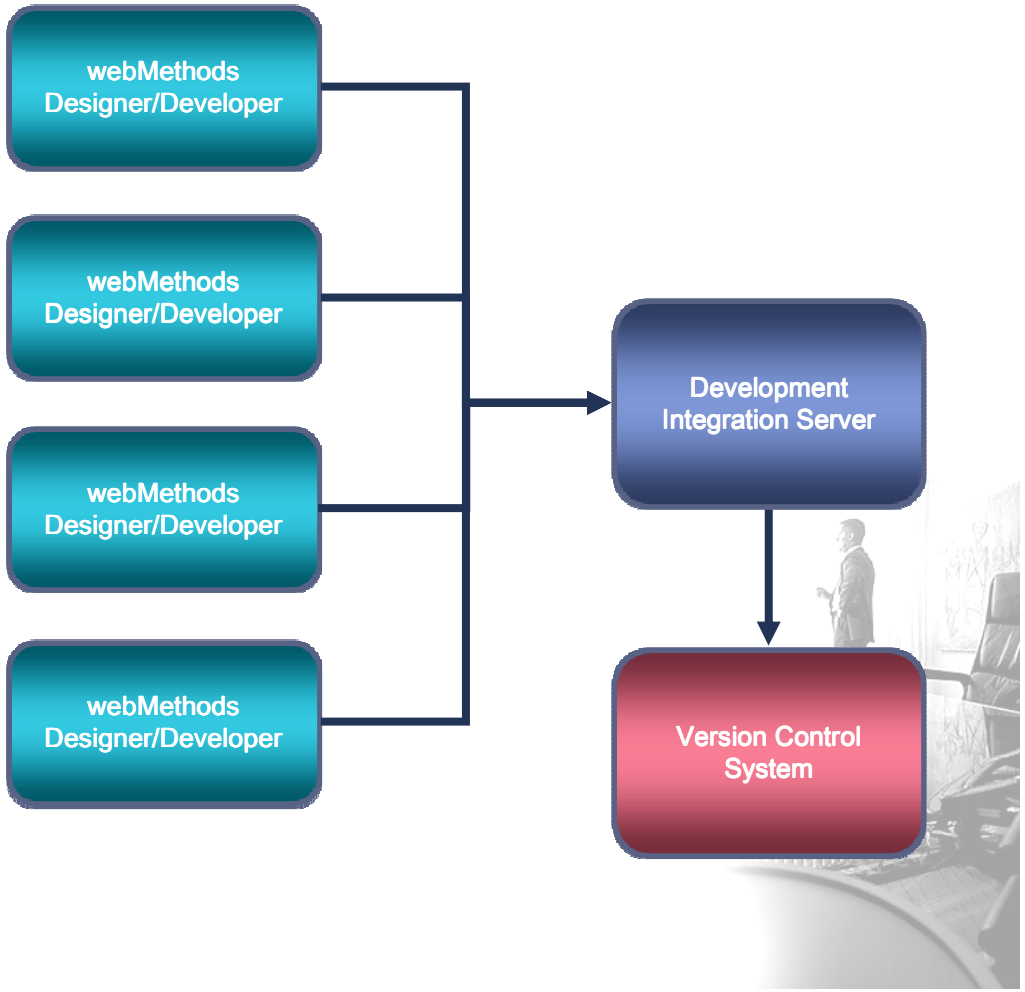
Agile Development: Traditional setup

PROS

- Out of the Box
- Easier installation
- No code compilation/ deployment
- Visible to all developers

CONS

- Limited VCS support
- Multiple versions/ releases difficult
- Need to be connected
- Java limitations



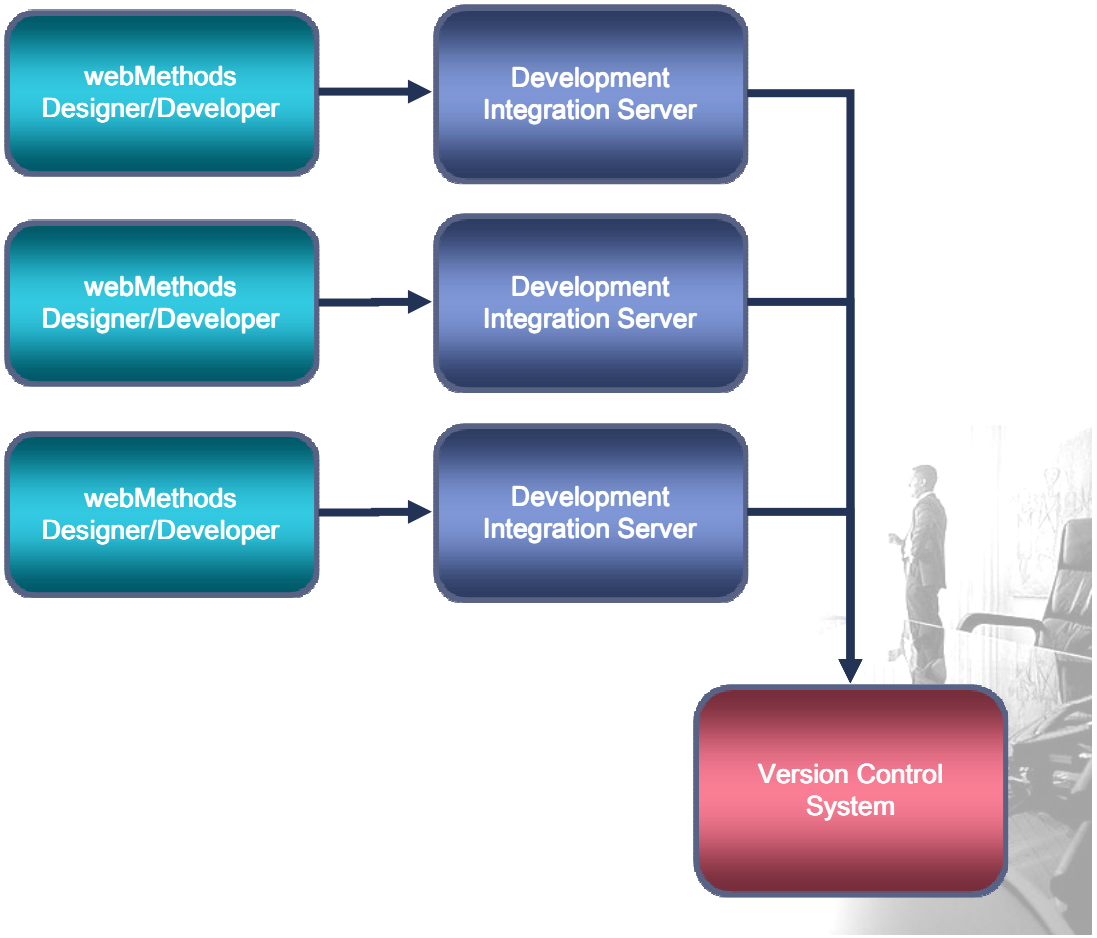
Agile Development: Local setup

PROS

- Developers get total access
- Designer/Eclipse plugins access
- Multiple releases
- Disconnected
- Better VCS integration
- No VCS product limitations

CONS

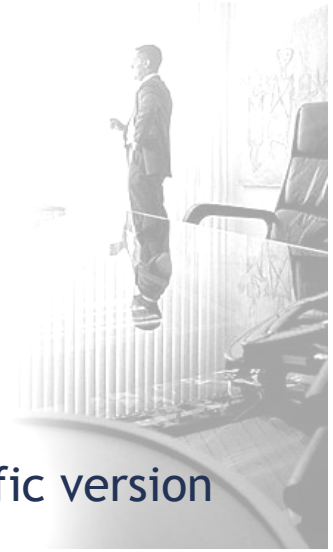
- Retraining
- Environment synch
- Licensing
- Dev Deployment needed



Agile Development: Local development

Important to have developers working in relative independence

- Product move towards local development
  - BPM/CAF development already local
  - Integration Server assets will be available in future release
- Consistent with development tools with develop, build, deploy cycle
- Testing challenges
  - Leads to more discipline in unit testing
- Environmental deployment and configuration challenges
  - Automation solutions improve and speed up processes
- Most important : Developer can ensure working with specific version



Agile Development: Local development

Relevant Tools

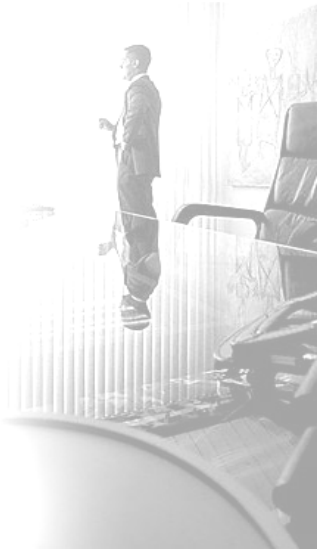
- Designer
  - Java development perspective
  - Project Builders
  - ANT

- Symbolic links
  - Connect Integration Server namespace to Designer workspace
  - Windows 7 : mklink
  - Junctions

- Designer VCS plugins

- WmBuildTools
  - GCS assets for local development
  - Automation of all activities

- GCS Local development white paper

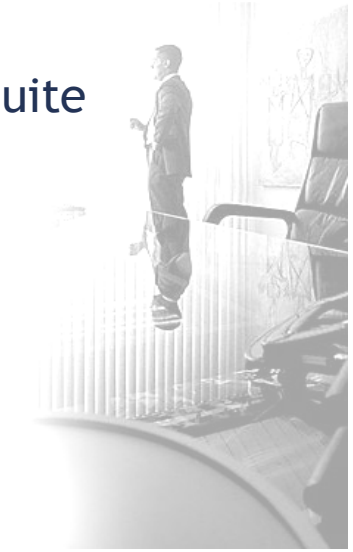


Agile Development: Test Driven

More moving parts, moving more frequently

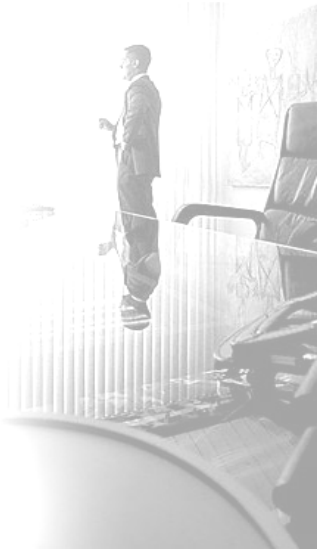
- Frequent releases
- More distributed development environments
- Need more testing
- Unit testing and regression testing
  - Leads to overall code quality
  - Core principal of agile – early detection of problems
  - Need framework and tools for unit testing – WmTestSuite

- Test automation
- Continuous Integration



Agile Development: WmTestSuite

- Unit and Regression testing
  - Based on JUnit, XMLUnit, HtppUnit, Xerces, Ant
- Codeless tests
  - Java developers can also use java code
- Integrated into Designer
  - Test Suite Design
  - Suite testing
  - Execution
- Backend and service calls can be mocked
- ANT integration
  - Execution & Report Generation



Agile Development: WmTestSuite

- Service Mocking
  - Eliminate backend system and data dependency
- Mock Types
  - Fixtured pipelines, Exception mocks, Service Mocks, Java code
- Mock scope
  - Session – affects current session only
  - User – affects current user
  - Server – affects all users
- Pipeline capture
  - Capture inputs/output for service replay
  - No code changes
  - Useful in regression tests, especially upgrades

Agile Development: WmTestSuite

wmTestSuite.xml

webMethods Test Suite

webMethods Suite

The Suite parameters are described here.

Name: Sample Tests

Description: Sample tests f

Filter:

webMethods Test Suite

The test cases are managed

More options, like executing

Duplicate Check Su
Duplicate Check Ca
Duplicate Check Su
Duplicate Check Su
mocks
wmServiceMocl
Duplicate Check Su
mocks
wmServiceMocl
Duplicate Check Su
Duplicate Check Su
mocks
wmServiceMocl

wmServiceMockSamples.services:getDuplicateLoans

Enter Input for 'getDuplicateLoans'

Include empty values for String Types

Name	Value
lienType	1
borrowerSSN	111-11-1111
propertyAddress	937 S Meyer
propertyZip	85701

Load Inputs | Save Inputs | OK | Cancel

Service I
Specify t
Service:
Invoke T
Browse...

Inputs
The inpu
File: re
Input

Expected
The expe
Type:
File:
Fields:
Output

wmServiceMockSamples.services:getDuplicateLoans

Output for 'getDuplicateLoans'

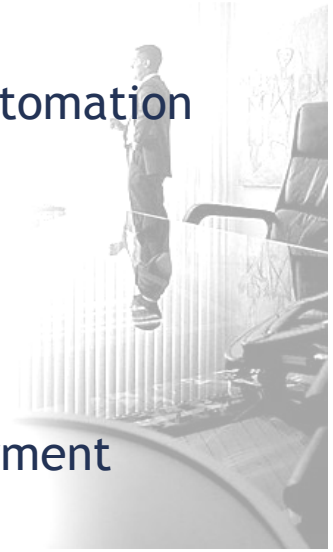
Include empty values for String Types

Name	Value
response	
@creationTime	/^2010-\d{2}-\d{2}T.*$/
duplicateLoans	
duplicateLoans[0]	
relevanceStatus	Relevant
loanNumber	0000000001
branch	TOTAL ADVANTEDGE LLC
underwritingDecisionExpirationDate	2005-04-27
lockDate	2005-12-20
lockIndicator	Y
tmoLoanStageDate	2005-04-27
product	C30
propertyAddress	937 S MEYER
propertyCity	TUCSON
propertyState	AZ
propertyZip	85701

Load Outputs | Save Outputs | OK | Cancel

Agile Development: Automation

- Human Errors contribute to most expensive problems
- Repetitive processes can be identified and automated
- Reduces process times
- Most relevant in testing and deployment
  - Large initiatives - complex testing and deployment scenarios
  - Better integration into build and deployment lifecycle
  - GCS framework and tools for automation
    - WmTestSuite: Test automation
    - WmBuildTools: Deployment, VCS integration & automation



- Automate what ?
  - Platform installation and deployment
  - Testing
  - Source Code Integration, Application Build and Deployment

Agile Development: Continuous Integration

Small and frequent quality control through regular builds and testing

- Maintain a Single Source Repository
- Automate the build and Make it self testing
- Everyone commits to the mainline everyday
- Everyone can see what's happening
- Automate deployment and test execution

Tools:

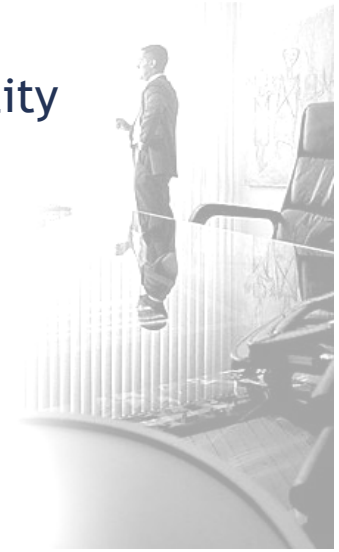
- CruiseControl, CruiseControl.NET
- Hudson/Jenkins
- Apache Continuum
- Team Foundation Server

Agile Development: Software Version Control

- Distributed Eclipse based development unlocks VCS potential
  - Better compare capabilities and merge certain assets
  - History
  - Tagging/Labeling/Branching
  - Pessimistic/Optimistic locking

- Software AG Designer moving in that direction

- Current Versions of some products don’t have that capability
  - GCS Tools bridge the gap
  - WmBuildTools
    - Aid local development
    - Deploy tagged version from VCS
    - Scripts for Continuous Integration



Agile Development: Distributed Deployment

Distributed development leads to Distributed Deployment

- Large scale integrated deployment
- Traceability: What code was in production when?
- Repeatable release/label/tag based deployment
- Ensures Consistency across environments
- Easier and faster switches require less environments

- Asset build environment : Version 8.2
- Older versions
  - WmBuildTools
    - Scripts for VCS extraction and build
    - Scripts to deploy tagged version from VCS
    - Scripts for test automation and Continuous Integration

System Landscape

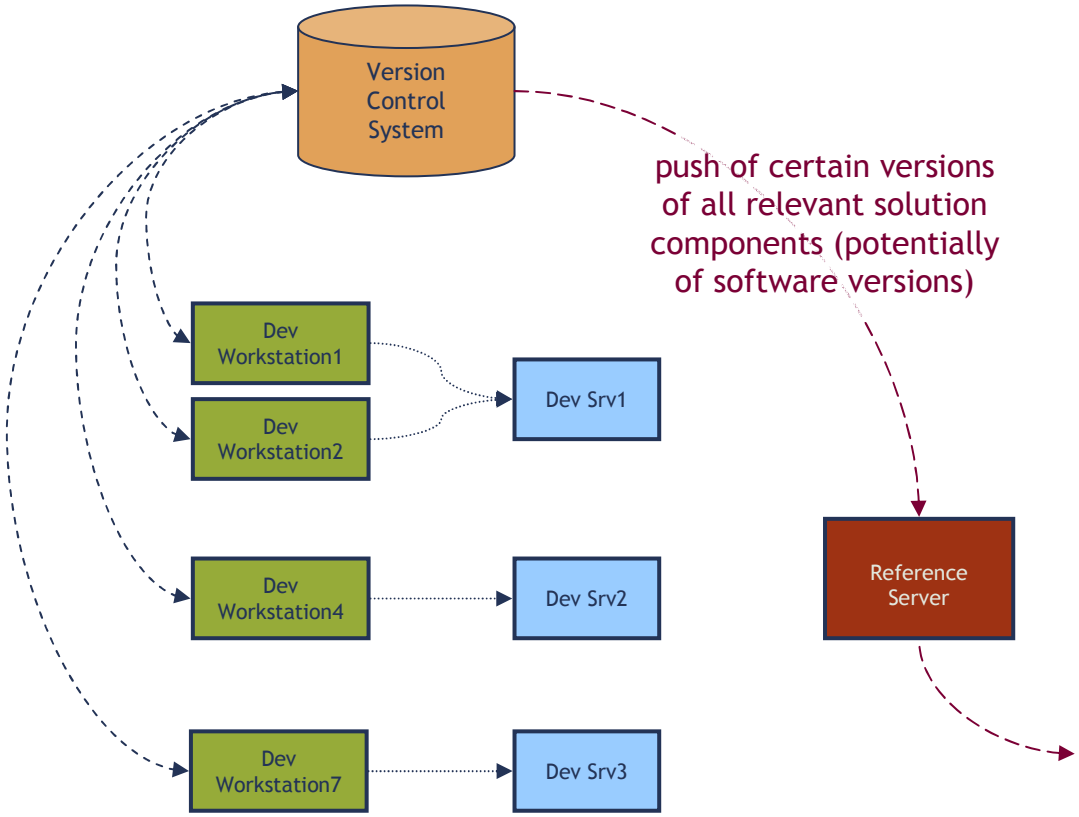
Source/Version Control System is the central repository for artifacts.

Multiple development servers for working on different projects or versions of the artifacts.

Reference Server is “populated” or “refreshed” from the artifact repository.

Development Servers can be partitioned based on projects, releases, types of users etc.

Development Workstations connect to the VCS either directly or indirectly in the case of Integration Server



Reference Server - Role

The term “Reference Server” is a concept. It may or may not be a single service or instance based on enterprise needs.

The main role of the Reference Server is :

- Use it as a basis for deployment into target environments
- Execute unit and regression tests against it
- Create a “dump” for auditing purposes
- Based on the actual setups, the Reference Server can also be used for Continuous Integration based testing.

- webMethods 8.2 Asset Build Environment

Integration Server – Build Process

- Checkout packages directly into the “packages” directory of Integration Server
- Re-compile and frag Java services using jcode
- Create custom configurations
- Create or update users, groups and ACLs
- Install or Activate package; usually required after first checkout (custom ANT tasks required)
- Synchronize publishable document type definitions to Broker (custom ANT task required)

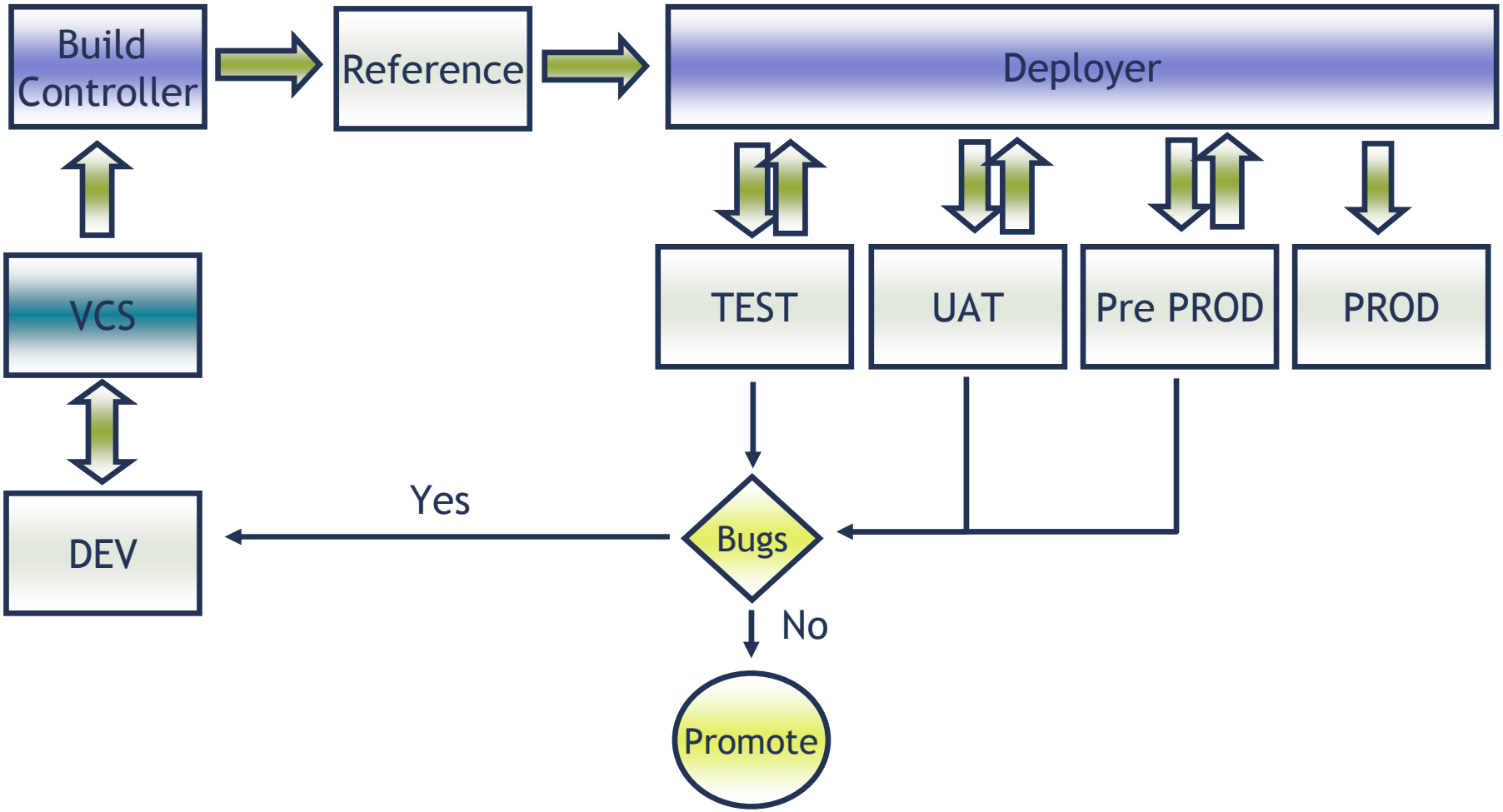
Integration Server - ANT Integration

- Build Process based on premise that ANT tasks can invoke services on the Integration Server.

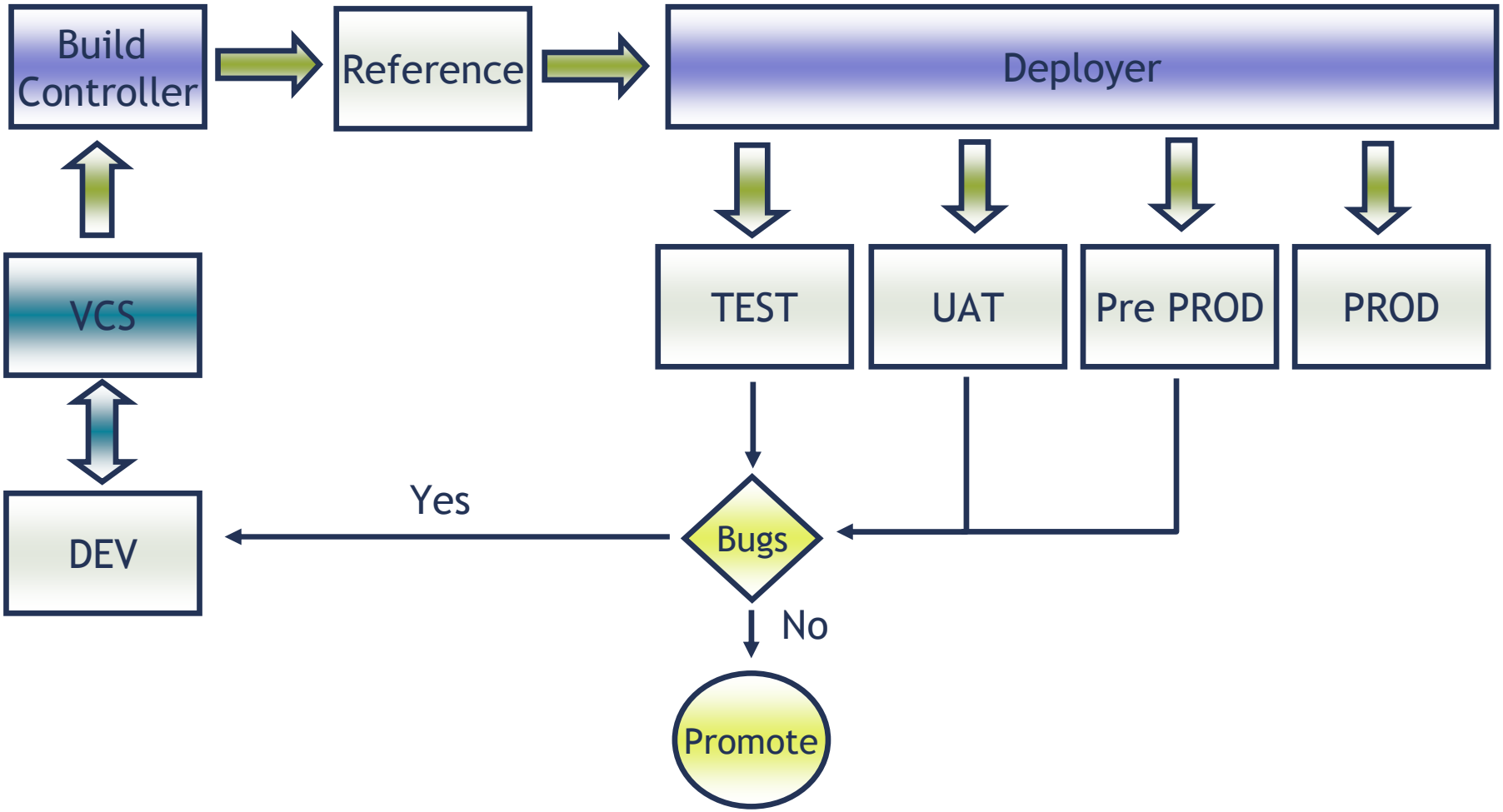
- Basic call pattern should be as under:

```
<wm:service folder="wm.server.access" service="aclAdd">
    <wm:idata>
        <wm:variable name="aclname" value="testAcl" />
        <wm:variable name="allowList" value="testGroup" array="true" />
    </wm:idata>
</wm:service>
```

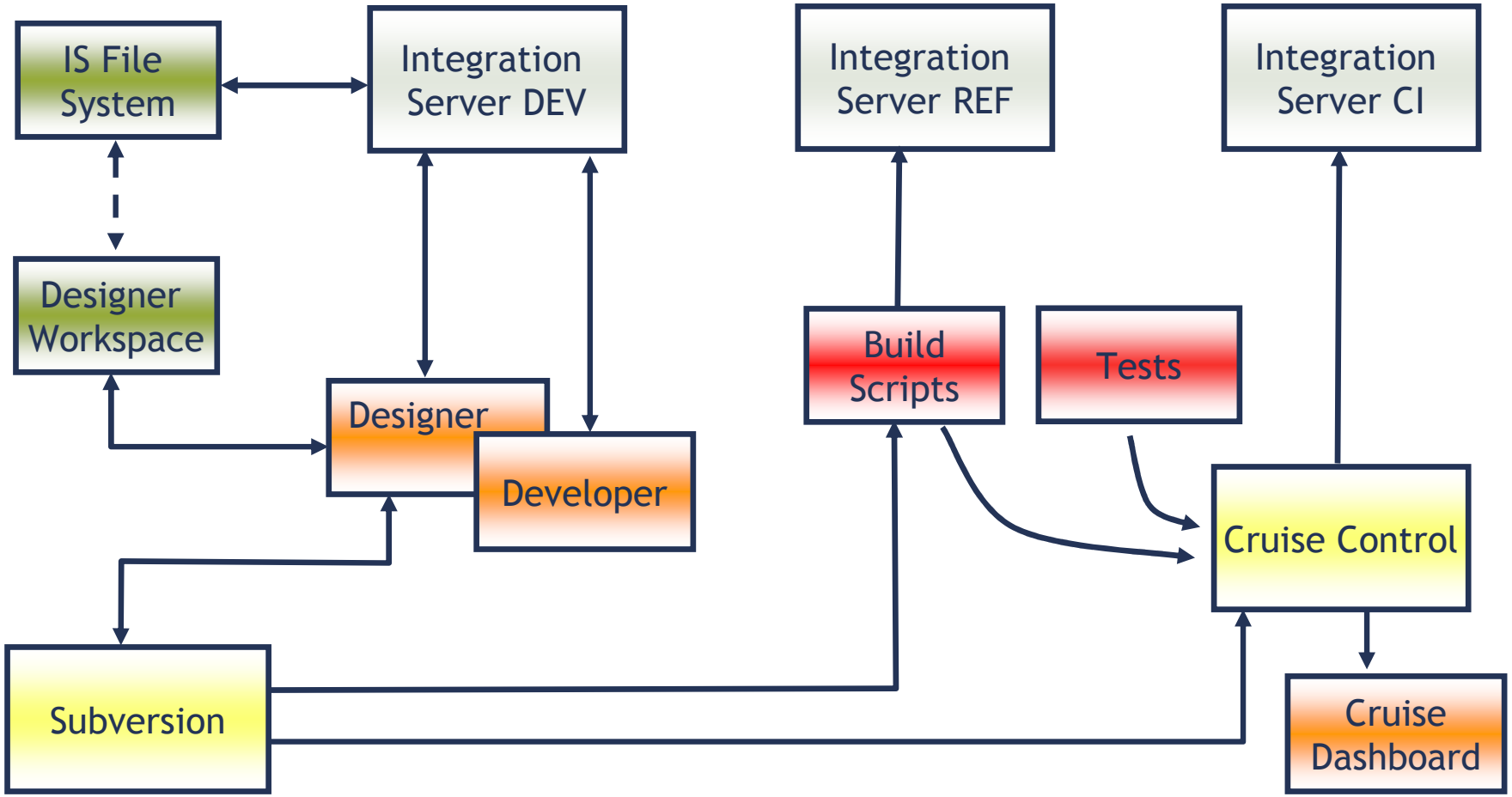
Staging or Asset Promotion



Staging or Asset Promotion - Alternate Path



Reference Development Architecture



software AG

Thank you!

PROCESSWORLD.2011