September 25, 2014

EFFECTIVE METHODS FOR SOFTWARE AND SYSTEMS INTEGRATION



PRESENTED BY:

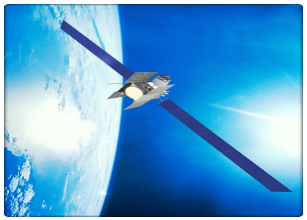
DR.BOYD L. SUMMERS

EFFECTIVE METHODS FOR SOFTWARE AND SYSTEMS INTEGRATION

- **Software Engineer (Quality)** **Defense and Space**

  The Boeing Company - Seattle, Washington



- **BL Summers Consulting. LLC** **"Software Solutions"**



OMICS Group
Accelerating Scientific Discovery

EFFECTIVE METHODS FOR SOFTWARE AND SYSTEMS INTEGRATION

Effective:

the implementation of achievable **schedules**, sound **processes**, and working **solutions** for software and systems integration.



* Stories about working in Integration Labs - Informal and Formal

Methods:

ensure **processes** and **tools** improve productivity and prepare for the **challenges** that impact integration environments.



Software:

design, code & unit test, plans, and test procedures integrated with applied systems, tell us that the software developed is done right.

"Peer Reviews" – KEY



* Management Meeting: HW and SW

Systems:

effort to accomplish and allocate **software design** and **engineering practices** to be defined and documented and be ready for the combination of software and systems integration activities. **"Teamwork"**



* Proactive and not Reactive Communication

Integration:

the compass to combine software, systems, firmware, and hardware to **work together** as **one**.



*Hardware Quality Assurance and Software Quality Assurance work together also.

Effective Software and Systems Integration methods provide an understanding and importance of critical factors for



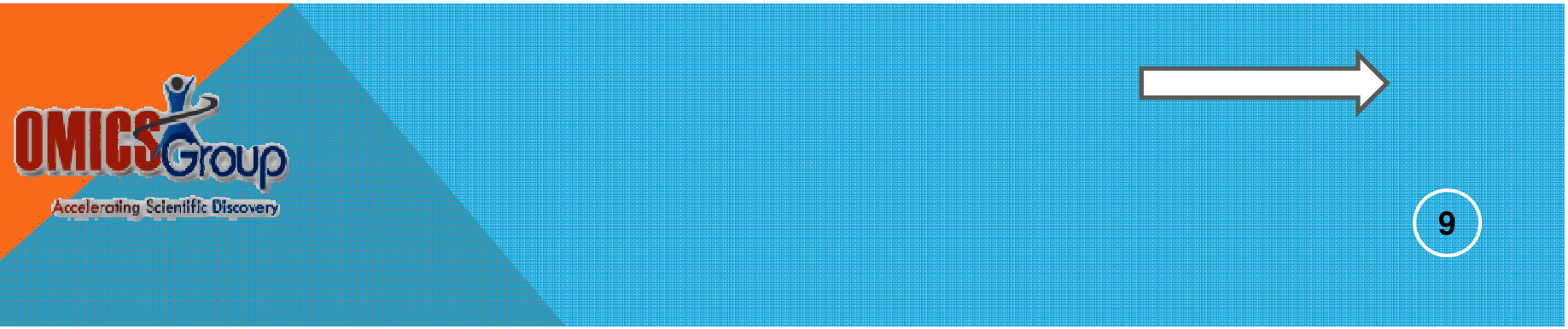
EFFECTIVE METHODS FOR SOFTWARE AND SYSTEMS INTEGRATION

Software and Systems Life-cycle

- planning
- systems design
- requirements
- software design
- configuration management
- integration testing
- quality
- customers

* Development life-cycles

* Key phases



EFFECTIVE METHODS FOR SOFTWARE AND SYSTEMS INTEGRATION

Critical understanding and the start of the right disciplines of integrated methods will empower and achieve effective, flexible, and quality results.

” Start the Right Disciplines ”

* Remember to always:



"Program and Project Planning"

The necessary steps to scope out systems and software design activities supporting integration. This type of planning will establish effective plans and results for:

- **communication**
- **risk management**
- **teamwork**
- **deployment**
- **quality before delivery**



"Systems Design"

The method to analyze requirements and develop a software development migration plan for defining:

- architecture concepts
- components and module development
- interfaces and necessary data

"Software Requirements"

Systematic approach for development of requirements derived from multiple resources:

- functional software interfaces
- plans, documentation, and procedures
- performance, verification and production

This method is applied for initial development of requirements and changes to baselines.

"Software Design/Development"

Systematic approach for creation of the design of software and the development to reflect design and definitions applicable to the work product defines details:

- work product architecture
- behavior and component interfaces

EFFECTIVE METHODS FOR SOFTWARE AND SYSTEMS INTEGRATION

A Software Manager interviewed a SW engineer for a new position in his company. He asked a question.

"How do you go about writing SW for a new project ?

The answer was "Start Coding the Software"

The Software Manager was expecting:

- Read the specifications, break down the software into individual modules and Flow Chart them, or something similar.

"He didn't get the Job" Why?



"Software Implementation"

Requirement for integration testing inside an integration facility provides assurance that engineering builds function as expected to enable smooth execution for verification and test activities.

* As software is tested and demonstration
  "lessons learned"

"Software Integration"

All software delivered or implemented by software integration or testing is processed through a configuration and controlled Software Library system, where status accounting of each delivery is maintained.

The integration tasks require that development and test processes are in place to ensure integration is ready for team troubleshooting and functional checkout.

"Software and Systems Integration"

Consistent approach to perform effective integration. Software units, components, and sub-systems are assembled within defined and documented plans and procedures to ensure software and systems elements are assembled properly.

The integration levels determine if constructed elements are ready for verification and/or validation activities.

Software and Systems Integration Delivery

Requires all testing to be performed and provide assurance that both software and systems are integrated and working together.

The integration practices ensure that units being tested are complete and documented prior to the official delivery or release to the customer.

"Product Evaluation"

An effective product evaluation method provides the necessary process steps to conduct and perform continuous evaluations of software work products during the development life-cycle and integration activities.

Evaluation tools and checklists are developed with associated scheduled processes to perform audits, reviews, and evaluations.

Quality - Software and Systems Integration activities *Software* and *Hardware* quality are required to support integration.

The test team run's through test installation procedures with the quality team looking over their shoulders to witness, verify and validate software to show that software works in a target system and results are documented for completion and closed.

Closing Statement

Effective Methods
for
Software and Systems Integration



Questions



