26th annual INCOSE international symposium

Edinburgh, UK

July 18 - 21, 2016

From Common Strategies and Approaches to

Virtual Integration

Moti Frank, HIT – Holon Institute of Technology

Uzi Orion, Elbit Systems Electro Optics – ElOp

Henry Broodney, IBM Research – Haifa

Sigal K. Kordova – Holon Institute of Technology

Introduction

- This presentation reviews the objectives of the integration process, planning and managing principles, as well as the pitfalls and difficulties associated with this process.

- Various integration approaches are described and analyzed.

Outline

- Systems Integration – Definitions
- Pitfalls and difficulties associated with integration
- Planning and Managing the Integration Process
- Integration Approaches:
  - Hardware-assisted versus software-only integration
  - Bottom-up versus top-down integration
  - Vertical (hierarchical) versus horizontal (functional) bottom-up integration
- Virtual Integration
- Integration Success
- Conclusion

Definitions

What is Systems Integration?

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

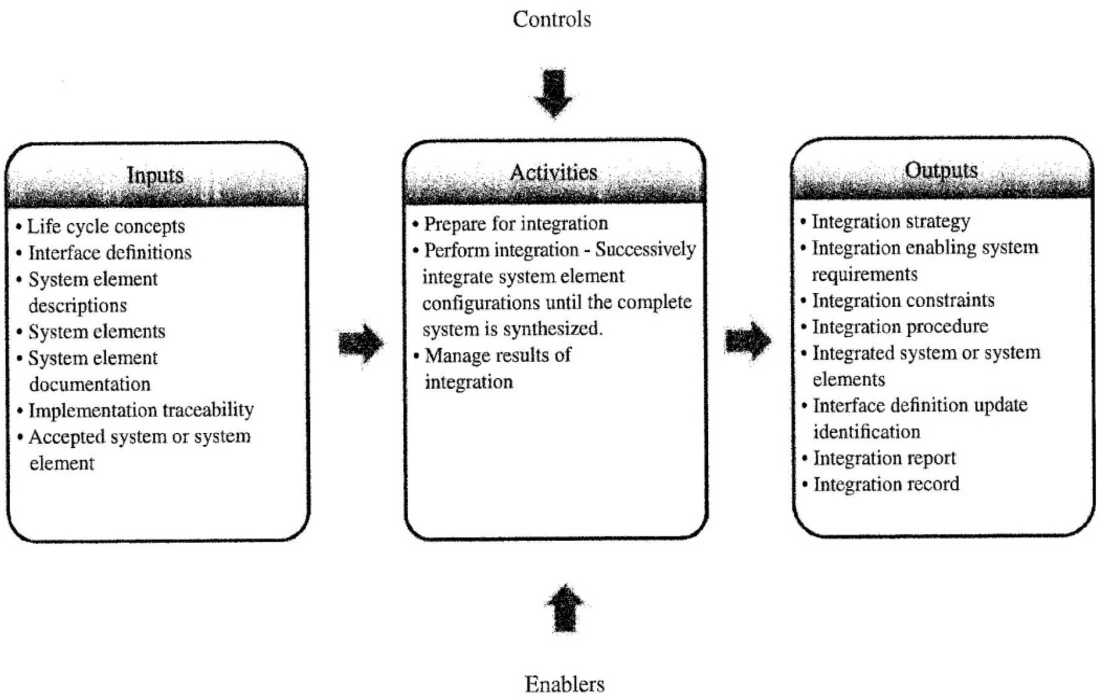
- Practices of joining a set of subsystems, hardware or software, to result in a single unified system that supports some need of the organization (Kuhn, 1990).
- Assembling of various hardware, software and human interfaces to accomplish a specific goal (Zaitun and Yaacob, 2000).
- Involves forcing the compatibility of subsystems so they work together (Westerman, 2001).
- A primary objective of systems engineering is to ensure the proper coordination and timely integration of all system elements (Blanchard and Fabrycky, 2005).

ISO/IEC/IEEE 15288 Definition

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- The purpose of the *integration process* is to synthesize a set of system elements into a realized system (product or service) that satisfies system requirements, architecture, and design.
- This process is iterated with the *verification and validation processes* as appropriate.
- The integration process includes activities to perform the integration of system elements (hardware/physical, software, human and procedures), verification and validation (V&V) test, and the demonstration of end-to-end operation (system build).
- The integration process addresses both the internal interfaces among the elements comprising the system and the external interfaces between the system and other systems.

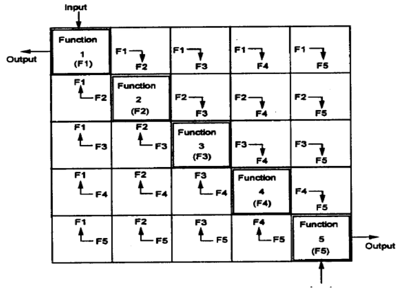
Input-Process-Output Diagram for the Integration Process
(INCOSE, SE Handbook, Fourth Edition, 2015)



The objectives of the integration process (Orion, 2014)

26th annual INCOSE international symposium
Edinburgh, UK

- verifying and validating the various parts, components or subsystems (referred herein as CI – configuration item);
- testing the interfaces between CIs;
- verifying the CIs' interoperability;
- identifying unexpected interactions between CIs;
- merging CIs into one entity, and testing that this entity meets the requirements and satisfies the customer's needs;
- identifying design deficiencies;
- for the first time, testing the system's operational and performance envelope, and verifying and validating the whole system in regard to meeting the requirements and satisfying the customer's needs.



Pitfalls and difficulties associated with integration

Pitfalls and difficulties associated with integration

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Djavanshir and Khorramshahgol (2007) have identified **fifty** key process areas of systems integration that should be well defined and addressed to ensure integration success.

- Bunza (1999) claims that “system **interfaces** are common sources of design, verification and test **problems** throughout the development process. Hardware-software interfaces are particularly troublesome”.

- Van Moll and Ammerlaan (2008) found that in more than fifty percent of development projects, the system integration phase was perceived by the project team as 'a problematic phase' or 'a phase that proceeded with severe difficulties'.

Pitfalls and difficulties associated with integration (cont.)

- Jain (2008) found that integration may be difficult and complicated for various reasons:
  - many disciplines and immature technologies are involved in the process.
  - key people often move to other projects (common in matrix organizations); as a result, they are not available to resolve problems associated with the integration process.
  - essential resources (test equipment, simulators, etc.) are not available on time
  - the participating parties are geographically far away
  - there are risks that were not mitigated in the projects' earlier stages.
  - unexpected interactions between parts and subsystems may emerge.

Pitfalls and difficulties associated with integration (cont.)

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Van Moll and Ammerlaan (2008), identified eight pitfalls related to the integration process:
  1. system integration and system integration testing are often distributed over several departments or parties without assigning overall responsibility
  2. system integration is often done at different physical locations, which makes it complex and difficult
  3. poor documentation and document management
  4. the responsible party is not an expert in all involved disciplines
  5. much time is lost on notifying the original developers and diagnosing the problems at hand
  6. delays in the delivery of subsystems
  7. the integration strategy is not well defined
  8. poor overall configuration management

Planning and Managing the Integration Process

Planning and Managing the Integration Process

(Blanchard and Fabrycky, 2005, Buede, 2009; INCOSE, 2010, Orion, 2011)

- Planning of the integration process should be done early in the development life-cycle.
- Usually, the strategic plan of the integration process should be performed from the system level down to the part level.
- However, the breakdown into integration paths should be done according to the integration logic, and not necessarily according to the system's physical hierarchy.
- The integration process should be planned in such a way that a single failure will not stop other integration paths.
- It is necessary to accomplish integration at a certain level before moving on to the next, higher, level.
- As far as is possible, it is recommended not to integrate more than two components at a time.

Planning and Managing the Integration Process (cont.)

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Before beginning the integration process, it is recommended to thoroughly test every component and every software module as a standalone unit.
- The integration process is dependent on the order of CIs readiness. If a particular CI is missing, then simulators or simulations might be used.
- Commercial-off-the-shelf (COTS) items should be considered as risky.
- Before entering any integration stage, it is important to identify the CIs that are required for this stage, engineering disciplines that will be involved, required resources and documents, and exit criteria.

Planning and Managing the Integration Process (cont.)

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- The prerequisite conditions for entering the next integration stage are:
  - All required CIs are available and have successfully passed all tests,
  - the configuration design for this stage has been checked and found to be correct,
  - all necessary infrastructure, equipment, tools, fixtures, documents, simulators, simulations, test equipment, chambers, power supplies, labs, instruments, test plans, work procedures, etc. are available.
- It is strongly recommended not to transfer unresolved problems to the next integration stage, as this simply exponentially increases the complexity of resolving problems in the next stage

Integration Approaches

1. Hardware-assisted versus software-only integration
2. Bottom-up versus top-down integration
3. Vertical (hierarchical) versus horizontal (functional) bottom-up integration

Hardware-assisted versus Software-only Integration (Bunza, 1999)

When dealing with hardware/software integration:

- Hardware-assisted methods include integration with external instrumentation, in-circuit emulator instrumented prototypes, hardware circuit emulators, and embedded monitors in the hardware prototype.

- Software approaches include simulators, simulations, and hardware/software simulation tools.

Bottom-Up versus Top-Down Integration

1. Bottom-Up Integration

- Bottom-up integration is commonly discussed in textbooks as the desired approach (Buede, 2009).

- In fact, the Vee model of systems engineering represents the bottom-up integration process as the appropriate one (Forsberg and Mooz,1991; Defense Acquisition University, 2001).

- The *bottom-up integration process* is usually described as a series of combining activities – first combining parts into components, next combining components into subsystems, and then combining multiple subsystems into the system.

- According to INCOSE (2010), "system build is bottom-up. That is, elements at the bottom of the system hierarchy are integrated and verified first".

Bottom-Up versus Top-Down Integration

1. Bottom-Up Integration (cont.)

The order of the **bottom-up** integration activities is usually as follows (Buede, 2009):

•Begin with a specific part, component or subsystem (referred to herein as CI – configuration item), inspect and test this CI, identify and fix any correctable deficiencies, assess the impact of any uncorrectable deficiencies, redesign the CI to address unacceptable impacts of any deficiencies, retest the CI and, if there are no unacceptable deficiencies, integrate with the next CI and test the new entity created by the integration of the two CIs. Repeat until all subsystems have been integrated.

•At each level, both the functional and physical combination of the parts, components and subsystems are tested and examined to determine if the appropriate outputs are obtained and whether the fit of these system elements is acceptable.

•It is not necessary to wait for the last available part or component before beginning the integration or before proceeding to the next level. In fact, simulations and models for a specific part, component or even subsystem are often used to reduce risk, speed up integration, and enhance testing efforts.

Bottom-Up versus Top-Down Integration
2. Top-Down integration

26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

- Commonly used in software engineering and appropriate for systems engineering.
- Top-down integration begins by assembling a model of the system built from simulations, simulators and models of the various CIs, and then testing the system's performance envelope.
- Once the modeled system is found to be verified and validated, the 'real' CIs are combined into the modeled system, as they become available and have completed testing.
- Gradually, all simulations, simulators and models are exchanged for 'real' units and the process ends when the real system is verified and validated.
- The main advantage of this approach is that early demonstration of the system is allowed, while the main disadvantage is that stubs have to be developed.
- This approach is most useful for systems using large amounts of commercial off-the-shelf (COTS) CIs (configuration items).

Vertical/Hierarchical vs. Horizontal/Functional Bottom-up Integration

1. Hierarchical (Vertical) Integration in SE

- A process in which the integration steps follow the physical hierarchy of the system.

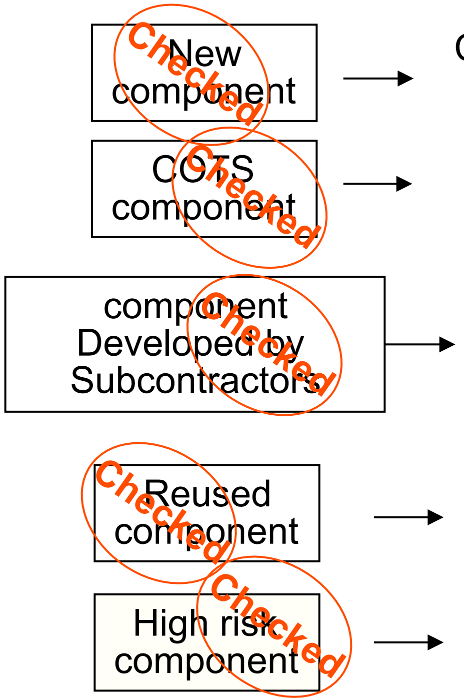
- This approach is commonly used when high risk is involved with the development of the various CIs.

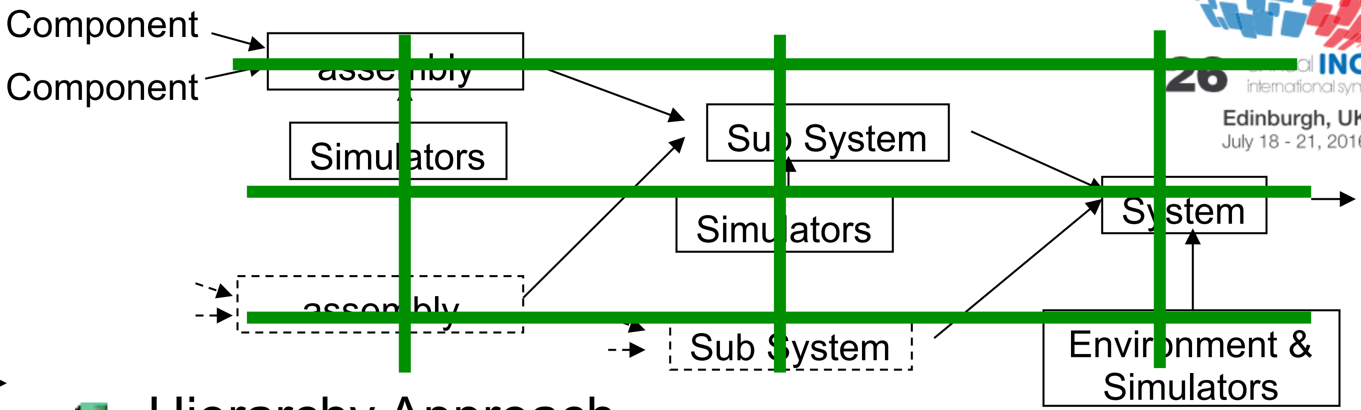
Vertical/Hierarchical vs. Horizontal/Functional Bottom-up Integration

2. Functional (Horizontal) Integration in SE

- A process of integrating subsystems according to their functionality by creating functional entities (Gold-Bernstein and Ruh, 2005).
- A process in which the integration steps follow functions of the system.
- This approach is commonly used when high risk is involved with some functions of the system. Partial functioning of the whole system is obtained at an early stage of the integration process.

Vertical/Hierarchical vs. Horizontal/Functional Bottom-up Integration





- **Hierarchy Approach**
  - Follows the logic of the system's physical hierarchy
  - Suits systems with risky physical components or technologies
- **Functional Approach**
  - Integrates complete functions of the system
  - Partial operation of the complete system is possible early in the integration process
  - Requires components completion for several assemblies to enable the process
  - Suits systems with risky functions

Vertical/Hierarchical vs. Horizontal/Functional Bottom-up Integration

3. The Integrated Approach

- The classification to *hierarchical* and *functional* is not dichotomous. Often, project managers choose the integrated approach, which combines the hierarchical and functional approaches.

- An *integrated approach* means that part of the integration paths are performed according to the principles of the hierarchical approach, while the other part of the integration paths are performed according to the functional approach.

Virtual Integration

Virtual Integration

What is?

- In business this term is used to describe the use of the Internet to replace physical components of a company with information. A business engaged in virtual integration owns only their brand and their clients. This eliminates the need to physically produce, ship or handle any products as they are now outsourced (http://www.businessdictionary.com).
- In supply-chain management this term represents the substitution of ownership with partnership by integrating a set of suppliers through information technology (IT) for tighter supply-chain collaboration (http://dl.acm.org).
- In SE, the term refers to the ability of examining the behavior and the performance of a system in a virtual environment.

Virtual Integration in SE

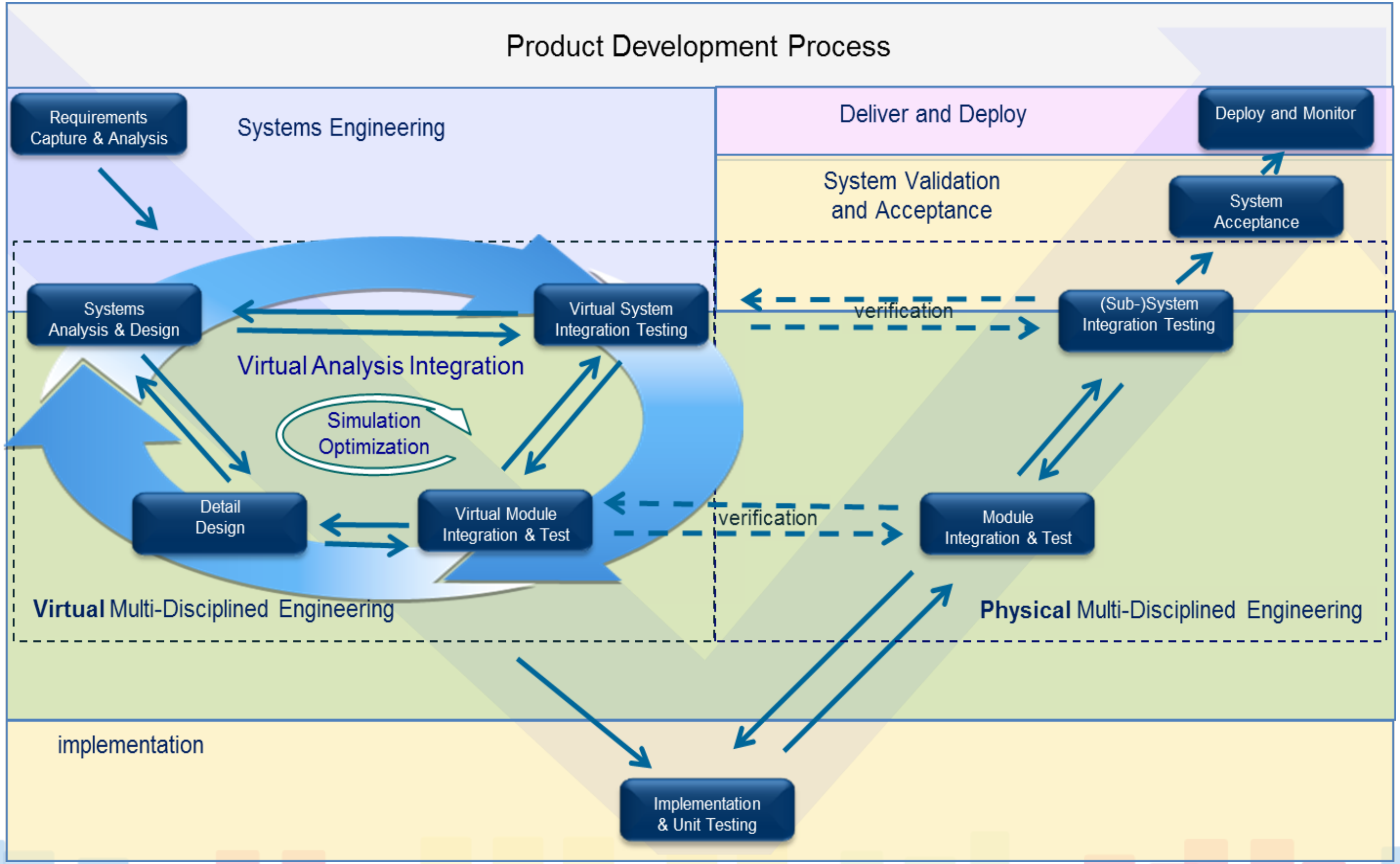
Main Principles

•Various fundamental technologies are used for Virtual Integration, such as formal verification, model checking, and the most common of all – simulation.

•System simulation is the most mature of all Virtual Integration technologies.

•since various system components' design progresses at different pace, at a certain point in the project there might be already physical implementations of some components, while others are still in the virtual phase. In such a case, approaches like Hardware-in-the-Loop can be applied.

•The process is driven entirely by Model-Based tools allowing cost effective exploration of design alternatives and evaluation of the whole system performance and correctness (see modified Vee-Model diagram next slide).



26th annual INCOSE international symposium
Edinburgh, UK
July 18 - 21, 2016

Integration Success

Integration Success

- There are many possible measures for assessing integration success.
- For example, the following measures might be considered:
  - number of identified design errors
  - architecture synthesis deficiencies
  - requirement incorrectness and inconsistencies
  - faulty interfaces
  - unexpected interactions between CIs
  - improper interoperabilities between Cis
  - system-level problems and faults
  - unexpected risks and failures in parts, components, subsystems and the whole system.
- Sometimes it is preferred to focus on the final result; that is, on project success rather than on measures related to intermediate stages of the project.

Conclusion

Conclusion

- The following issues were discussed:
  - Definitions of Systems Integration
  - Pitfalls and difficulties associated with integration
  - Planning and Managing the Integration Process
  - Integration Approaches:
    - Hardware-assisted versus software-only integration
    - Bottom-up versus top-down integration
    - Vertical (hierarchical) versus horizontal (functional) bottom-up integration
  - Virtual Integration
  - Integration Success

Thank You!
Q & A