ETLS 509 - Validation & Verification
University of St. Thomas

John Engelman

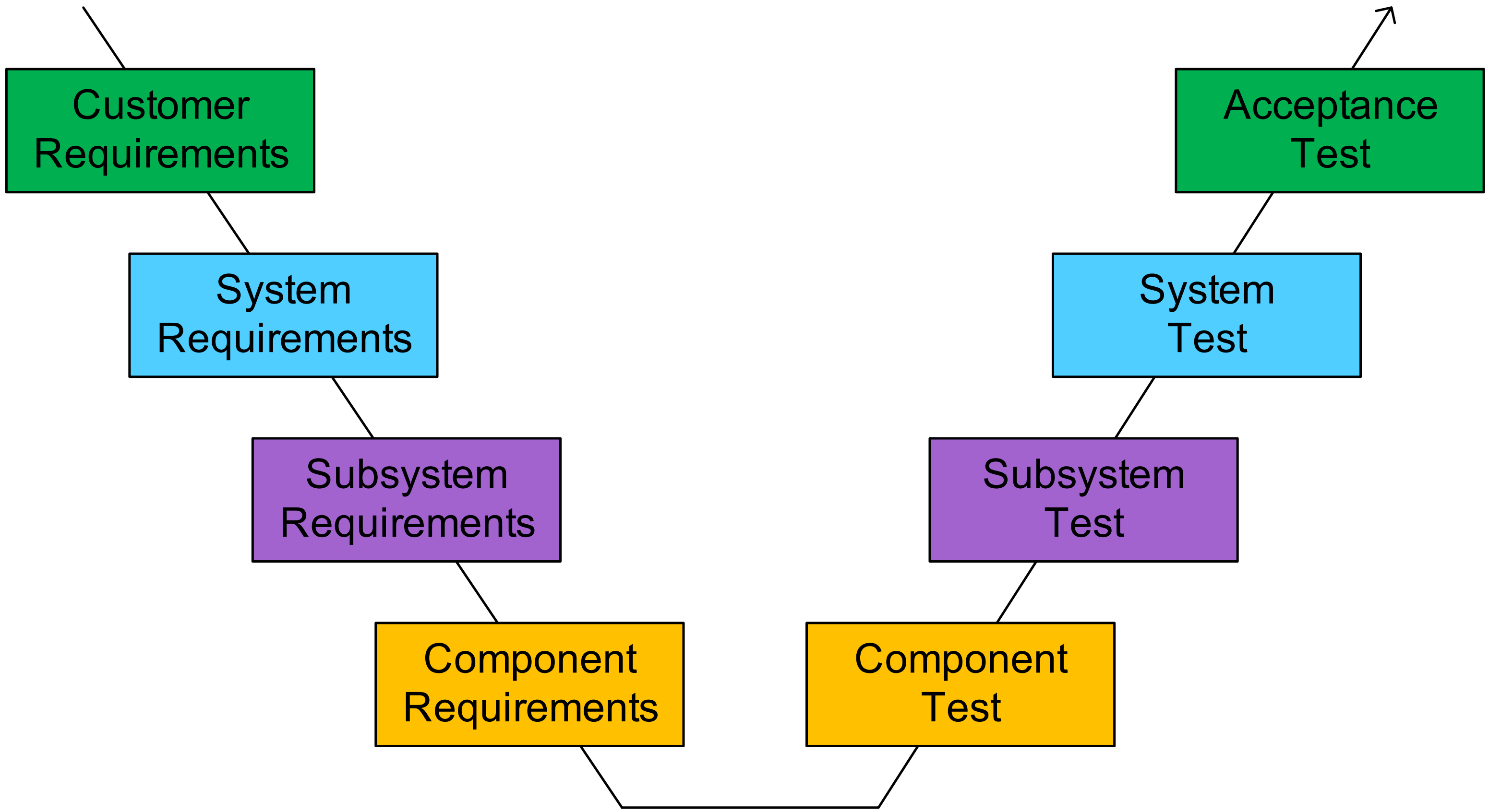
Fall 2016

ETLS 509 - Session 5

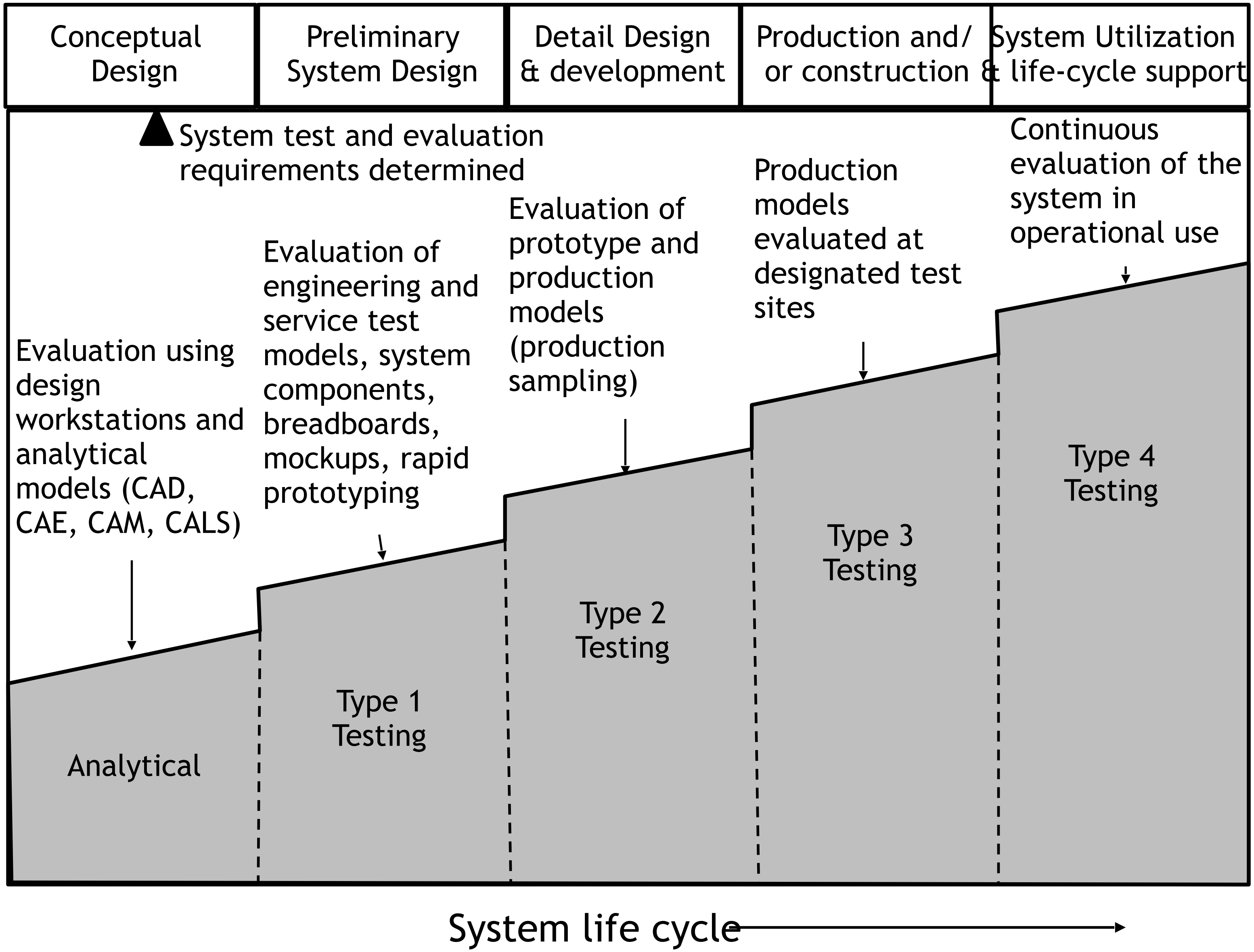
- Question/Review
- Systems V diagram,

ETLS 509 - Session 5

- **System Test & Evaluation – Type 3 & 4 testing**



Stages of Evaluation During the Life Cycle[1]



Note: “type 1, 2, 3, 4” testing is a Blanchard designation

1 Blanchard fig. 6.2

Testing During Stages of a System's Life Cycle

- **Conceptual design**
  - Analytical – evaluation using design workstations and analytical models. Normally things have not been built. There may be items for considered for use in the design that are tested by other methods (unusual).
- **Preliminary design**
  - Engineering & service test models, Breadboards, mockups
  - Basically testing of potential key items related to the design
- **Detailed Design & Development**
  - Bulk of design testing occurs at this stage
  - Performance tests, Environmental Qualification, Structural, Reliability Qualification, Maintainability, Support Equipment, Personal Test and Evaluation, Technical Data Verification, Software Verification
  - This is "Blanchard's Type 2" testing
- **Production**
  - Production models evaluated at designated test sites
  - Formal Factory Acceptance Tests (FATs) – This may include a subset of the tests done during qualification testing, e.g., temperature cycling may be required.
- **System Utilization & Life Cycle Support**
  - Operations tests ensuring proper operation of the system
  - May include subsets of the FATs or specialized test developed for use in operational environment

Detail Design & Development Testing

- **Bulk of system development testing occurs during this phase of a systems life cycle**
  - Component testing to verify components are functioning correctly, nominal and “end case” conditions
  - Subsystem verification testing is done just as component testing is done to verify correct operation
  - Development testing concludes with system qualification testing

Note – Blanchard takes the view that System Test occurs as part of “Type 3” testing occurring during production. I’ve found more commonly that full system test & qualification is required prior to going into production. System evaluation is something that begins during the development stage and must typically pass a number of performance related milestones prior to being authorized for production.

System Test & Evaluation

- **Formal tests conducted after initial system qualification, prior to construction.**
- **Conducted by user personnel at operational test site.**
- **Operational test & support, operational software, spares**

Note – Blanchard takes the view that System Test occurs as part of "Type 3" testing occurring during production. I've found more commonly that full system test & qualification is required prior to going into production. System evaluation is something that begins during the development stage and must typically pass a number of performance related milestones prior to being authorized for production.

Testing During the Utilization Phase of a System

- **Testing that is done as part of the operation of a system**
  - Consider taking off in an airplane, part of this consists of mechanics/pilots going through an extensive checklist to verify proper functionality of the aircraft. A number of these items will be tests that are performed to verify the system is responding as expected.
- **Formal tests into specific areas**
- **System capability is "tested" via operational use of the system**
- **First true test of system capability**
- **Additionally, formal testing accomplished in conjunction with technology enhancements, systems upgrades**

System Test, Data Collection, Reporting & Feedback

System Assumed to be in Operational Use

- General system operational and support factors
- Operational and maintenance software
- Operational and maintenance facilities
- Transportation and handling
- Personnel and training
- Supply system (spares and repair parts)
- Test and support equipment
- Technical data and information handling
- Consumer (user) response

Testing Considerations

- **Typically testing of all conditions is not feasible**
  - Consider you're responsible for testing a processor chip that performs double precision multiplications
    - Double precision number is 64 bits, multiplying two double precision number yields a total of $2*2^{64}$ possible combinations of input numbers or a total of $3.7*10^{19}$ possible combinations
      - If a test of multiplying 2 number could be done in one nanosecond (unlikely), it would take something over a 1000 years to complete the test
  - Understanding how the system and its components are put together is essential to building an effective set of tests and test plan

Edge Testing

- **Edge testing is testing that occurs with a parameter of the item being tested that is at its maximum or minimum value**
  - If a system is intended to operate in an environment from 0^oC to 50^oC, then edge testing of the temperature refers to the testing done a 0^oC and 50^oC
  - As a system is put together, “edge testing” of parameters is something that is commonly done, e.g.,
    - Components or subsystems will be tested at maximum and minimum temperatures, voltages, etc.
    - Software will have test cases that include variables with maximum and minimum input values
  - Subsystem testing that is done only under nominal conditions is a recipe for errors in the overall system.

Corner Testing

- **A corner test of a system is a test where each parameter that is variable for a system is at either its maximum or minimum value**
- **If there are n parameters that define the conditions under which a system operates then there are a total of 2^n possible “corners”**
  - Suppose an electrical motor is being tested in a subsystem, what are the potential variables
    - Temperature
    - Humidity
    - Voltage
    - Load
      - initial
      - On-going
  - In this simplified case, there are 16 corners reflecting system start-up and 16 corners reflecting on-going use.

Corner Testing - continued

- **In most systems there are likely to be a large number of parameters that are variable**
- **In general it is not feasible to test all corners the system**
  - If there are a 100 different parameters, there are 2^{100} or over a billion different corner conditions.
- **There will be a subset of parameters for a system that need to be verified at their maximum/minimum values**
  - For some parameters, it may only be necessary to test one edge of the parameter with the system otherwise operating under “nominal” conditions
    - This is especially true if subsystem test have the parameters covering a wider range

Choosing Corner/Edge Tests

- **As the number of parameters is almost certainly going to be too large to enable complete corner testing choosing test conditions is critical to the overall success of the system meeting its operational goals**
  - Extensive testing is typically done with parameters under what are considered "nominal conditions"
  - Where things usually go wrong, baring component failure, is areas that have not been tested
    - These conditions frequently involve one or more parameters at the edge of their input values with other parameters being closer to nominal
- **Based on a system design, it is more likely to be sensitive to some parameters versus others**
  - In developing the test plan, the sensitivity of the system to given parameters must be considered. The edges of these parameters that are most likely to cause issue are those that need to be explicitly called out as part of the testing

A Real Example

- **At one point in time Xilinx produced an FPGA (field programmable gate array) XCV1000E FG680 that was specified to be able to operate from -40ºC to +100ºC**
  - The maximum current draw at -40ºC was specified to be 2 amps.
  - Problems in applications utilizing the FPGA were found where the FPGA was drawing more than 8 amps on power-up and some devices would not power-up at all
    - In the cold temperature test that Xilinx did on the extended temperature devices the test had limited the current to 2 amps, with the current limited to 2 amps, the devices powered-up just fine. When the current limitation was removed from the test at cold temperature a significant percentage of the FPGAs failed the test.
    - It was a customer test engineer that identified the problem with the Xilinx test
  - The factory acceptance testing of the device did not reflect the use, according to Xilinx specifications, in an operational environment.
  - Discovering that this was a fundamental failing in Xilinx acceptance testing was exceptionally expensive time, product recall from those using the Xilinx part, as well as the extensive engineering effort required to find the problem

Destructive Testing

- **Destructive Testing -**
  - Testing a system or component under operating conditions until the system or component fails
    - Stretching a spring back and forth until the spring no longer meets it required specifications
  - A test that destroys the system or component as a result of the test
- **Why Destructive Testing**
  - Testing lifespan of a component, subsystem, or system
  - Testing portions of a system that are only used when things are destroyed
    - 777 episode 32:00-34:10
    - Auto crash testing
      - Purpose of the test is to determine the effectiveness of the auto’s subsystems that are to protect the passenger in the event of a crash

Auto crash test example

- **Videos:**
  - 2014 Chevrolet Spark
    - http://www.youtube.com/watch?v=h0rr6nIzm50&list=PLyFBJQcOqvQGp1EP-5dD8itgeZgGZa8Oc&index=5
  - 2013 Smart electric drive
    - http://www.youtube.com/watch?v=le1NKv8ijyE&list=PLyFBJQcOqvQEgRGObVkWIEJQBjKbIzz5R&index=40

Ford Focus Crash Test

Destructive Test Results Matter
- Crash test result meanings from DOT HS 811 359

Rating	Meaning - Side Impact
*****	5 percent or less chance of serious injury
****	6 percent to 10 percent chance of serious injury
***	11 percent to 20 percent chance of serious injury
**	21 percent to 25 percent chance of serious injury
*	26 percent or greater chance of serious injury

Rating	Meaning - Frontal crash
*****	10 percent or less chance of serious injury
****	11 percent to 20 percent chance of serious injury
***	21 percent to 35 percent chance of serious injury
**	36 percent to 45 percent chance of serious injury
*	46 percent or greater chance of serious injury

Accelerated Life Testing

- **Life testing is performed to determine the life span, typically the number of hours of operation of a system or component**
  - Unfortunately, from a life testing perspective, the number of hours of operation of a system is likely to dramatically exceed any reasonable schedule
- **Accelerated life testing**
  - Is intended to determine the life span of a component by subjecting it to conditions that significantly exceed the normal operating conditions
    - Voltage, temperature, vibration, pressure, stress, strain as well as potentially other conditions
  - The key to accelerated life testing is to understand what the effect of the "acceleration factors" on the system/ component

Accelerated Life Testing - continued

- **A historical example (acceleration factors are no longer correct)**
  - In the 1980's with the semiconductor processes that were used temperature could be used as an acceleration factor for life testing
    - Typical rule of thumb was that for every 10^oC you increase the operating temperature, the life span was reduced by a factor of 2
    - If the normal operating temperature was 40ºC by subjecting the component to 120^oC the acceleration factor would be 2^8 or a factor of 256. As such a 168 hour test (one week) would provide information on a 43000 hour life span
- **Accelerated life testing is a major issue in system design to verify the product reliability/availability**
  - Acceleration factors are determined by the types of items that make up a system

Time/Budget Constraints

- **System testing is constrained by schedule (time) and budget**
  - There is virtually no limit to how long a system could be tested or the amount of effort expended in testing
  - Ideally we would like to say for X amount of time (schedule) and Y dollars, the produced system will be error free
    - Not possible from a practical perspective or even theoretically
      - It is straight forward to show that completeness of testing is a mathematically unsolvable problem
  - Boeing 777 – episode 3 15:00-21:45
- **What approaches are taken**
  - Knowledge of the system gives an understanding of parameters for which the system is likely to be more sensitive
    - System loading
    - Temperature if the system has electrical/mechanical components
    - Many others that are a function of the type of system
- **Customer required testing**
  - In some cases a customer may explicitly require a certain level or certain types of tests
  - These need to be built into the overall test plan regardless of other testing that needs to be done to ensure proper system functionality
  - Frequently the tests are implicitly required versus being explicitly required, e.g., a requirement to show the system can tolerate a given amount of radiation (e.g., satellite)
    - How do you do this type of inherently destructive testing without using up system life?

Final thoughts on testing

- **You can never have to much data about how your system is performing**
  - The fundamental question is can you utilize that data that is collected
- **Well thought out testing will develop a base of data for continuing improvements**
  - E.g., six sigma programs
- **Remember that your competition is also out there, looking at your product as a basis for their next improvements (the bigger, better deal):**
  - mega hertz competition in computer chip development
  - mega pixel competition in digital cameras
  - feature competition in the auto business
  - All these items must go back to the “improvements” on the solution to the problem being solved.

Boeing 777

- What are the what Testing / Type 3 / 4 testing considerations for the 777?
- 21st Century Jet - Making the Boeing 777 6-4
- 21st Century Jet - Building the Boeing 777 - Full Episode 5 00_02_00-00_06_00.mp4