

IT Methodology Webinar



Leverage the Software Development Lifecycle

PROJECTinsight®

Project & Portfolio Management Software

Initiate Project Intelligence®

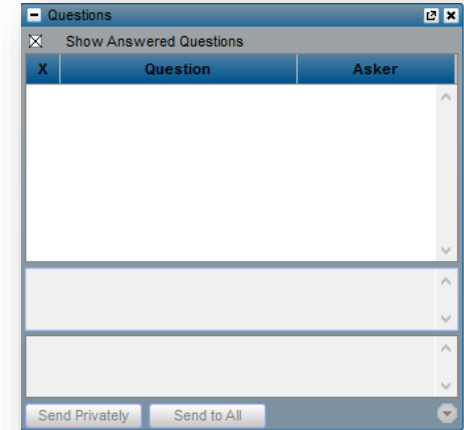
Things to Know...



- ▶ All participants will be on mute
- ▶ Questions are welcome
- ▶ Ask questions in the question box
- ▶ We **DO NOT** send out PDU certificates
- ▶ May be valid for PDU credit (**self-report session**)

- For more information check out article on Webinars and PDUs

<http://www.projectinsight.net/support/kb/webinars-and-pdus>





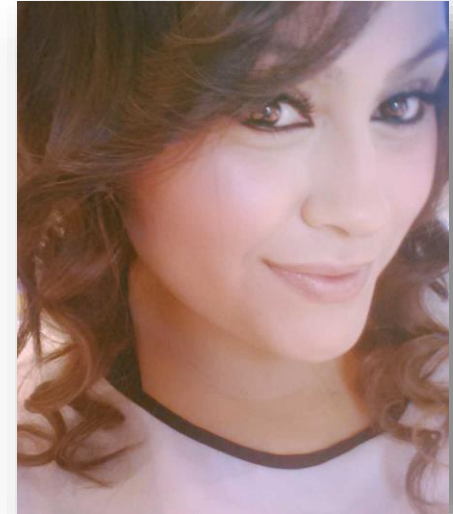
Denise Rodriguez

Project Insight

Marketing

Denise.Rodriguez@projectinsight.com

www.projectinsight.net





Schedule a customized demo today!

- +1 (949) 476-6499 x3
- info@projectinsight.net
- Request info: www.projectinsight.net

Presenter



Cameron Watson

President, QAIassist

cwatson@qaiassist.com

www.qaiassist.com



Who's on the call?



- ▶ I am a Project Manager
- ▶ I am a Business or Operational Manager
- ▶ I am a software developer or software tester
- ▶ I am a member of an organizational PMO, governance or quality assurance function
- ▶ I want a better understanding of IT Methodology

Who is QAlassist?



- Established 2007 – Ottawa, Canada
- Nurturing organizational efficiency through IT Methodology knowledge & expertise
- Support and promote a community with evolving IT Methodology knowledge and expertise
- Maintain, optimize, promote and support the delivery of IT methodology products and lifecycles proven to increase operational performance
- Administration and oversight of QAlassist Integrated Methodology (QAlassist-IM) and formal certifications – “Foundation” & “Practitioner”
- Support practitioners and authorized affiliates in acquiring and delivering IT Methodology knowledge and expertise



IT Methodology Webinar Audience



● **Nurturing efficiency through IT Methodology knowledge & expertise across the globe**

- Over 200 countries
- Wide array of industries (manufacturing, banking, health, insurance, consulting)
- More than 50 unique webinars

● **Over 10,000 webinar registrants**

- Executives
- Operational & Delivery Managers
- Project Managers
- Business Analysts and Architects
- Application Developers & Testers
- Corporate PMO, Quality Assurance, Continuous Improvement
- Consultants



QAlassist 2016 Series - Goals



- Share IT Methodology knowledge
- Develop IT methodology understanding & expertise
- Provide additional insight into several IT (project management, software development, software testing) methodologies
- Discuss best practices on incorporating an organizational IT methodology
- Promote ongoing dialog and feedback with webinar audience (questions, email, survey)

2016 Schedule



- **The 3rd Wednesday of Every Month**

- 8:00 am Pacific - 11:00 am Eastern Time

- **IT Methodology Concepts (60 mins)**

- This series of six webinars will be presented on a bi-monthly basis. Each webinar will provide you with a context, overview, terminology and general understanding of information technology (IT) methodology.

- **IT Methodology – Tutorials (45 mins)**

- This series of five webinars will be presented on a bi-monthly basis. Each session is designed to offer a more “hands on” perspective – bridging the concepts into a practical utilization.



2016 Concept Sessions



- ▶ **The 3rd Wednesday of Month (Jan, Mar, May, Jul, Sept, Nov)**

- 8:00 am Pacific - 11:00 am Eastern Time

- ▶ **IT Methodology Concepts (60 mins)**

- **A Context for IT Methodology - (Jan)**
- **Apply PM Fundamentals to IT - (Mar)**
- **Leverage the Software Development Lifecycle - (May)**
- **Explore the Software Testing Lifecycle - (Jul)**
- **Incorporate Organizational Process Governance - (Sept)**
- **Implementing IT Methodology - (Nov)**

2016 Tutorial Sessions



- **The 3rd Wednesday of Month (Feb, Apr, Jun, Aug, Oct)**
 - 8:00 am Pacific - 11:00 am Eastern Time
- **IT Methodology Tutorials (60 mins)**
 - **Applying IT Methodology to Project Initiation - (Feb)**
 - **Project Planning & Design with IT in Mind - (Apr)**
 - Executing Projects with IT Methodologies - (Jun)
 - Project Control & Verification - (Aug)
 - Project Close & Delivery - (Oct)

2016 Concept Sessions



- ▶ **The 3rd Wednesday of Month (Jan, Mar, May, Jul, Sept, Nov)**

- 8:00 am Pacific - 11:00 am Eastern Time

- ▶ **IT Methodology Concepts (60 mins)**

- A Context for IT Methodology - (Jan)
- Apply PM Fundamentals to IT - (Mar)
- **Leverage the Software Development Lifecycle - (May)**
- Explore the Software Testing Lifecycle - (Jul)
- Incorporate Organizational Process Governance - (Sept)
- Implementing IT Methodology - (Nov)

Leverage the Software Development Lifecycle Concepts III



► Webinar Goals

- Review previous IT Methodology Concepts Sessions (Jan, Mar)
- Define a context for Software Development Lifecycle (SDLC)
- Discuss significance of Requirements and Traceability
- Identify the purpose and objectives of SDLC phases

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Leverage the Software Development Lifecycle

Concepts III - Agenda



- **Establish webinar context - review previous “Concepts” sessions**
 - **Jan - A Context for IT Methodology**
 - **Mar- Apply PM Fundamentals to IT**
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Terms, Terms and more Terms



▶ IT Framework, IT Methodology, IT Lifecycle

- Non Standard usage in terminology
- Terms are used interchangeably between organizations and across the IT industry
- Examples include
 - Project Management Methodology/Lifecycle/Framework
 - Software Development Methodology/Lifecycle (SDLC)/Framework
 - Software Testing Methodology/Lifecycle/Framework
- **Every IT organization is using an IT Methodology or IT Lifecycle or IT Framework**

IT Methodology Context



- Organizational Frameworks
 - ISO(900XXX), CMMI, Six-Sigma, ITIL, COBIT
- Methodologies/Lifecycles (noun)
 - Rational Unified Process (RUP), QAlassist (IM), PMI, Prince2
- Methodologies/Lifecycles (verb)
 - Waterfall, Spiral, Agile, RAD, etc

IT Methodology Context



- **Organizational Frameworks**
 - ISO(900XXX), CMMI, Six-Sigma, ITIL, COBIT
- **Methodologies/Lifecycles (noun)**
 - Rational Unified Process (RUP), QAlassist (IM), PMI, Prince2
- **Methodologies/Lifecycles (verb)**
 - Waterfall, Spiral, Agile, RAD, etc

IT Methodology – Map & Journey



Methodologies/Lifecycles - “noun”

- The roadmap
- Rational Unified Process (RUP), QAlassist (IM) , PMI, Prince2

Methodologies/Lifecycles - “verb”

- The journey taken along the road
- Waterfall, Spiral, Agile, RAD , Prototyping, etc

IT Methodology/Lifecycle - Noun



- ▶ All Methodologies/Lifecycles (RUP, QAIassist-IM, PMI, Prince2) define, utilize and rely on their own pre-defined structure, naming conventions, and terminology
- ▶ Hierarchy/Structure
 - Methodology/Lifecycle (ie Project Management, Software Development, Software Testing)
 - Phase/Stage (ie Initiate, Planning, Design, Closeout)
 - Deliverable/Artifact/Work Product (ie Project Charter, Testing Strategy)
 - Activity (tasks performed to complete deliverables, artifacts, work products)

QAlassist Integrated Methodology



QAlassist Integrated Methodology

| | Initiate | Plan | Execute & Control | | Closeout |
|----------------------|---|---|--|---|--|
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Project Closeout Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| | Systems Analysis | Design | Build | Test | Release |
| Software Development | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

“Deliverable/Artifact/Work Product” versus “Activity”



Building an IT Application

● Deliverable (Artifact, Work Product)

- Business Case
- Project Plan
- User Acceptance Test Plan

● Activity (non-Deliverable, non-Artifact, non Work Product)

- Reviewing business need with Operational Management, Business Analysts and End-Users
- Meeting with Stakeholders to acquire approval of Project Plan
- Discussing User Acceptance Testing objectives and priorities with testing team



► Fixed Sections (always populated)

- Title Page
- Deliverable Details (Author, Creation Date, Version, Status, etc)
- Deliverable History, Reviewers, Sign Off
- Table of Contents (as per deliverable)
- Context (purpose of the deliverable)

► Variable Sections (based on pre-defined informational requirements)

- Structure/format defined as per deliverable (ie a Project Charter deliverable does not have the same pre-defined informational requirements as a Testing Strategy deliverable)

Deliverable/Artifact/Work Product Status



- **Draft** - Deliverable has been identified, assigned and is in the process of being drafted/completed
- **Review** - “Draft” deliverable has been completed and submitted for “review” and “authorization”
- **Approved/Archived** - “Reviewed” deliverable has been “Authorized” by designated stakeholders and put under configuration management
- **Applied** - Information within “Authorized” deliverable is being referenced and/or utilized within another deliverable

“Deliverable/Artifact/Work Product” Attributes



● **Tangible (pre-defined informational requirements)**

- Can be identified and defined (completion dates, cost) on Project Schedule
- Can be assigned (populating information requirements) to project team member(s)
- Can be referenced by project team members to support creation of other deliverables
- Can be formally reviewed and officially authorized
- Will alter in status (authored, reviewed, approved, archived, referenced) throughout life of project
- Will reflect evolving deliverable and project status/progress (risks, issues, percentage complete)
- Can be placed under formal configuration management – re: with evolving status
- Can be accessed for future reference (application support and maintenance)

Methodology Fundamentals 101



- Viable IT methodologies (noun and verb) are designed to help not hinder
- IT methodologies (noun) are applicable for a wide range of IT Methodologies (verb) and delivery approaches (waterfall, spiral, agile, RAD, prototyping, etc)
- No hard or specific rules (ie iterations, frequency of iterations, deliverables to be completed) of how an IT methodology (noun) can best be applied or leveraged by an organization or project team
- No silver bullet - if the perfect combination of IT Methodology (noun) and IT Methodology (verb) existed, every project team and organization would be using it

Methodology Fundamentals 102



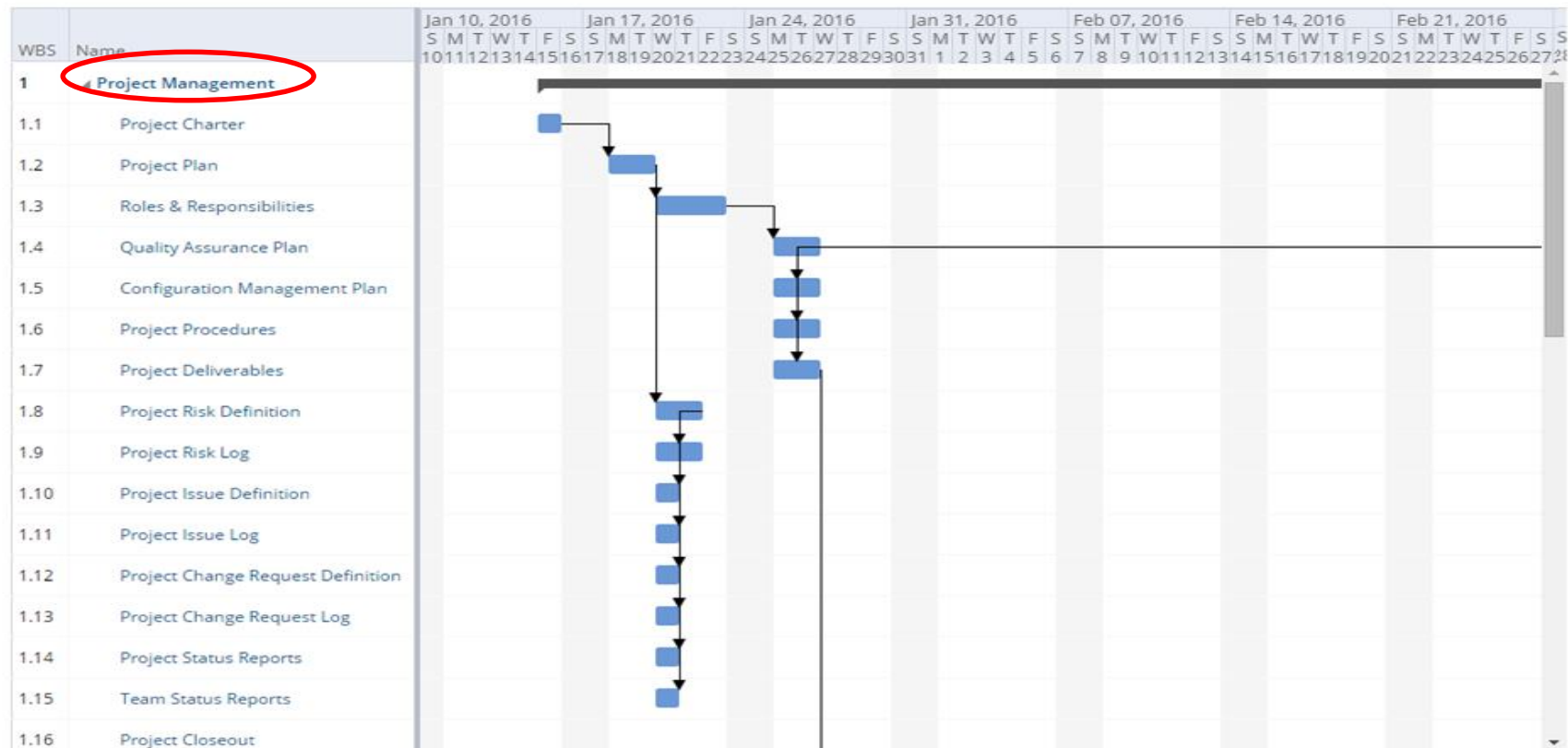
- Deliverables (and content) may be re-usable from one project to another
- Not every deliverable of every phase has to be completed for every project
 - any and every deliverable to be completed must add value (cost, schedule, quality) to the project
- Not every section or sub-section of every deliverable has to be completed for every project - information used to populate every deliverable must add value (cost, schedule, quality) to the project

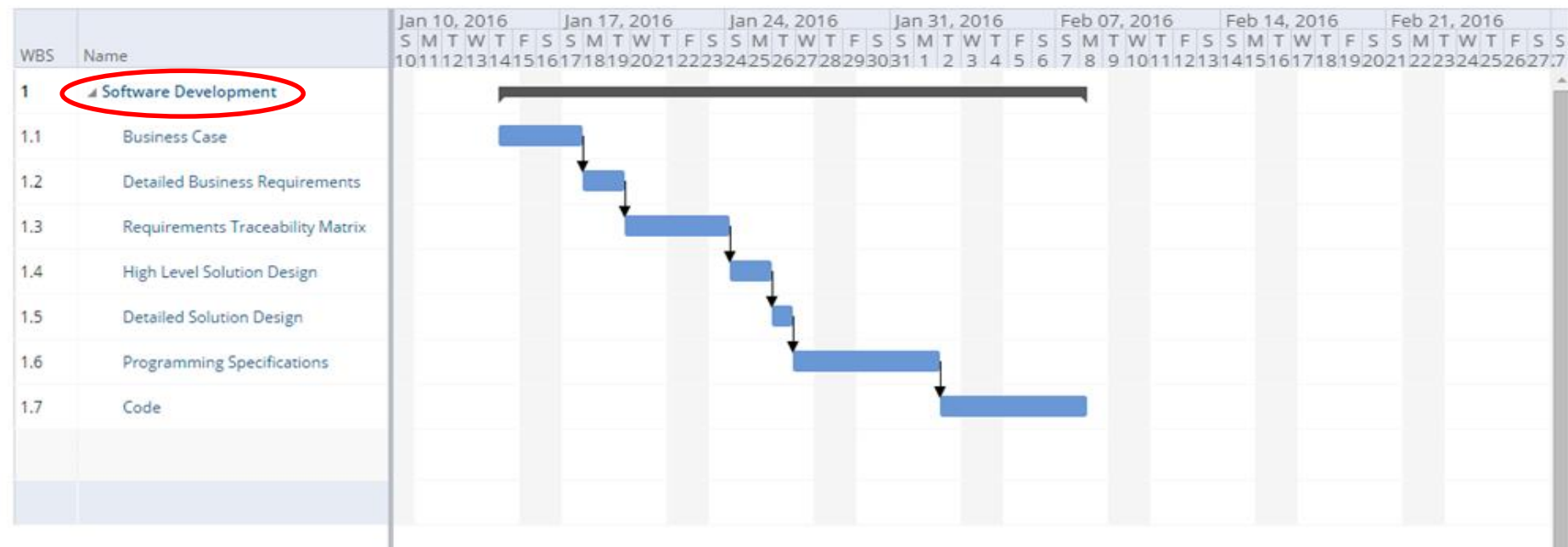
QAlassist Integrated Methodology

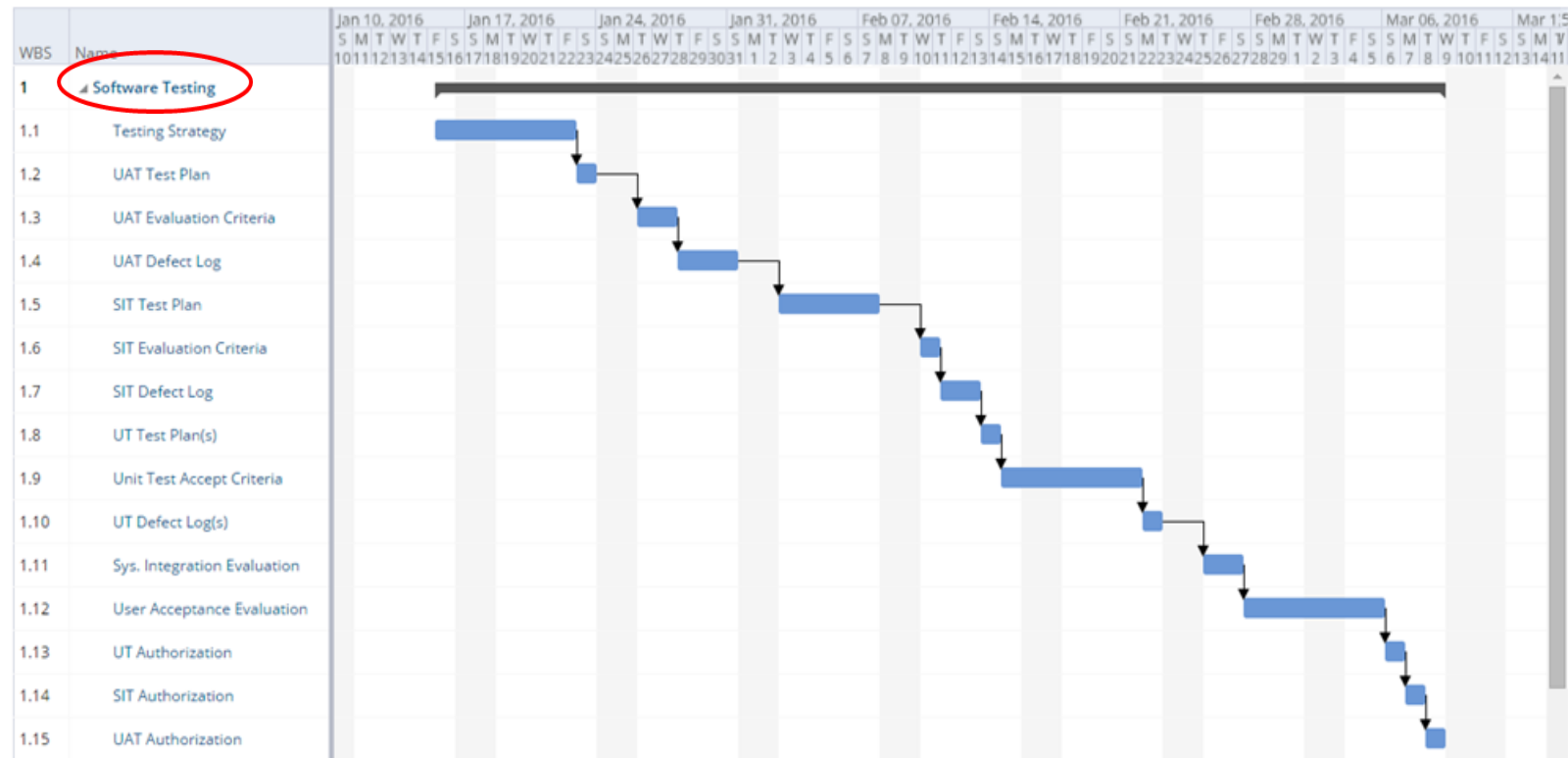


QAlassist Integrated Methodology

| | Initiate | Plan | Execute & Control | | Closeout |
|----------------------|---|---|--|---|--|
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Project Closeout Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| | Systems Analysis | Design | Build | Test | Release |
| Software Development | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |







Leverage the Software Development Lifecycle

Concepts III - Agenda



- **Establish webinar context - review previous “Concepts” sessions**
 - **Jan - A Context for IT Methodology**
 - **Mar- Apply PM Fundamentals to IT**
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- **Software Development Lifecycle (SDLC) – Context and Premise**
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap



► Development Environment

- New Products
- New Operational Business Applications (ie HR, Accounting, Sales, etc)

► Maintenance/Support Environment

- Existing Products
- Existing Operational Business Applications (HR, Accounting, Sales, etc)
- Off the Shelf Applications

SDLC – IT Context



- ▶ Multiple IT Environments
 - Mainframe, Distributed, Web Based
- ▶ Multiple Delivery Approaches – Methodology (verb)
 - Waterfall
 - Agile
 - Prototyping
- ▶ Integrations
 - Project Management Lifecycle
 - Software Testing Lifecycle

SDLC – Inherent Principle



- ▶ How many requirements (and associated functionality) should/can be developed ?
 - Too Few – leads to a solution that does not satisfy the business need
 - Too Many – leads to a solution that is over priced, over built, over budget and past due (and may still not satisfy the business need)

How can a project team know when they are best leveraging an
SDLC to develop the appropriate level of functionality ?



Identifying, Designing, Building and Unit Testing functionality

**** Functionality cannot be delivered in a vacuum ***

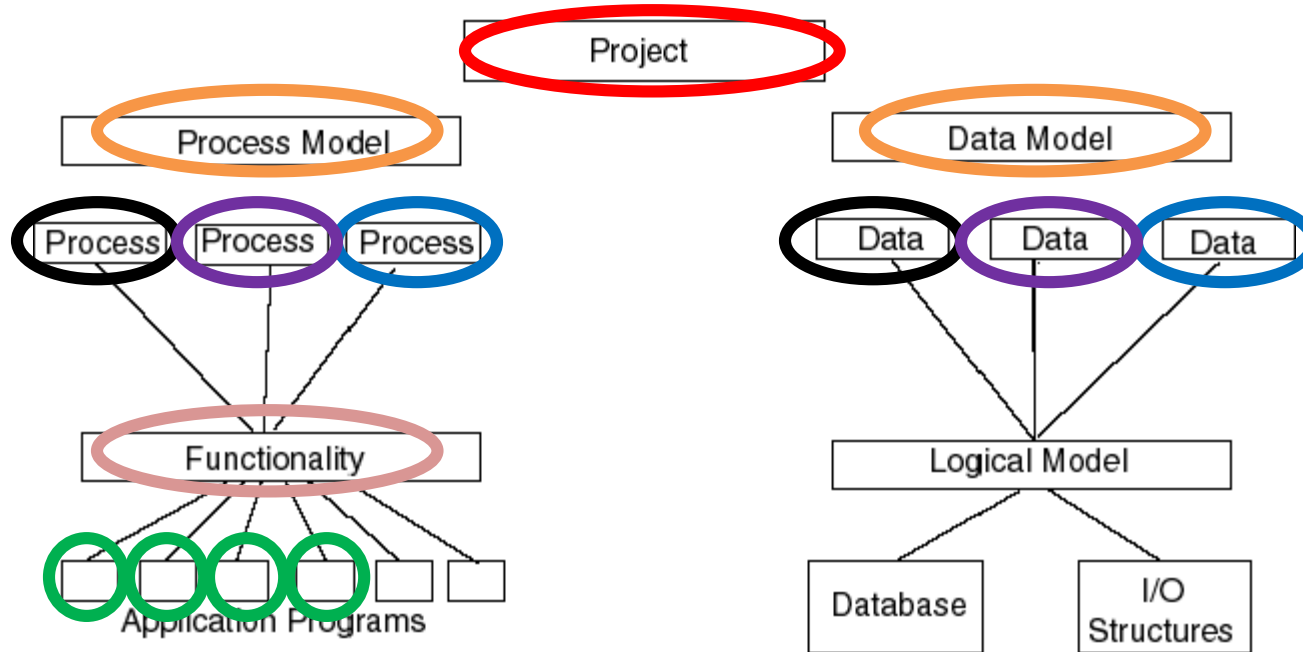
- **Manual Functionality**

- Processes that deliver or administer the delivery of products and/or services (often leverages automated functionality)

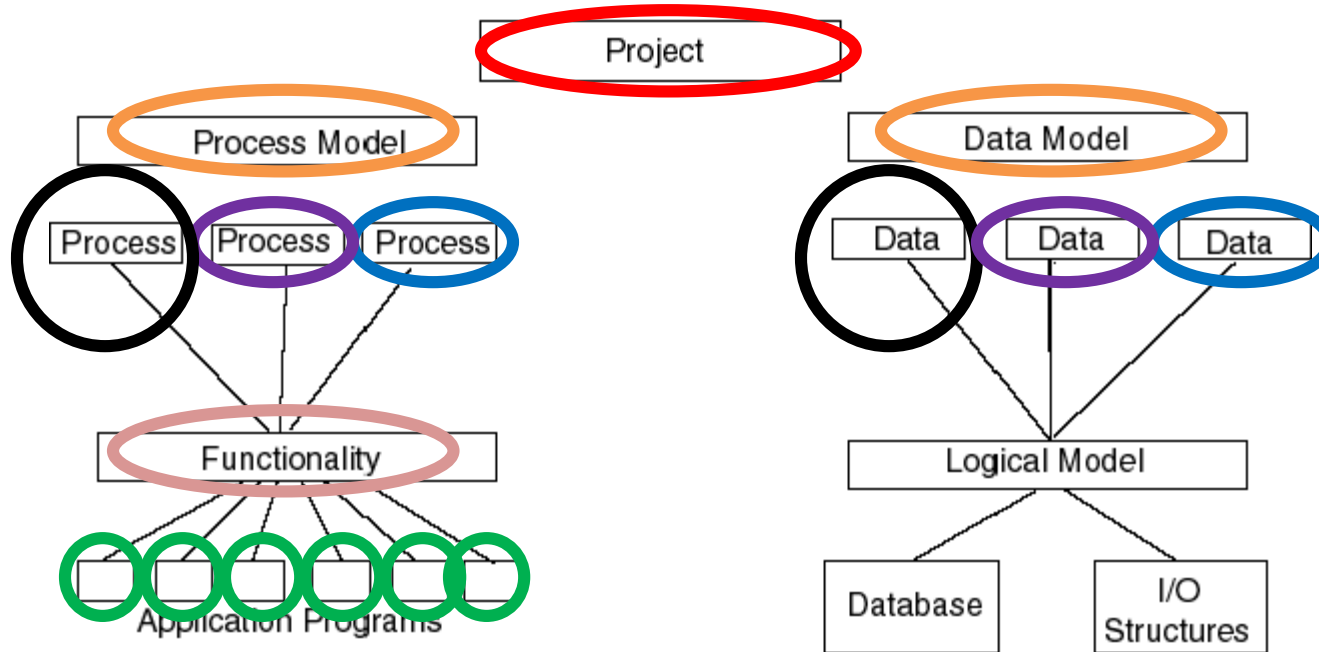
- **Automated Functionality**

- IT Systems/Applications that interface and support the delivery of **Manual Functionality**

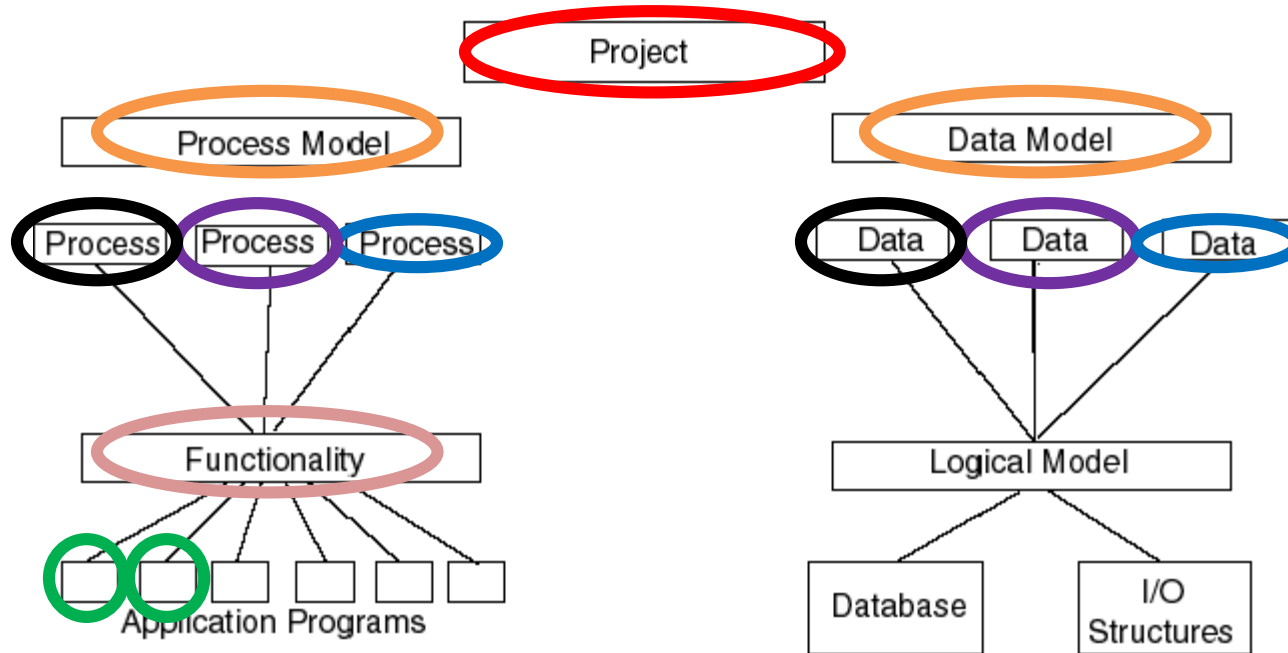
SDLC – Premise



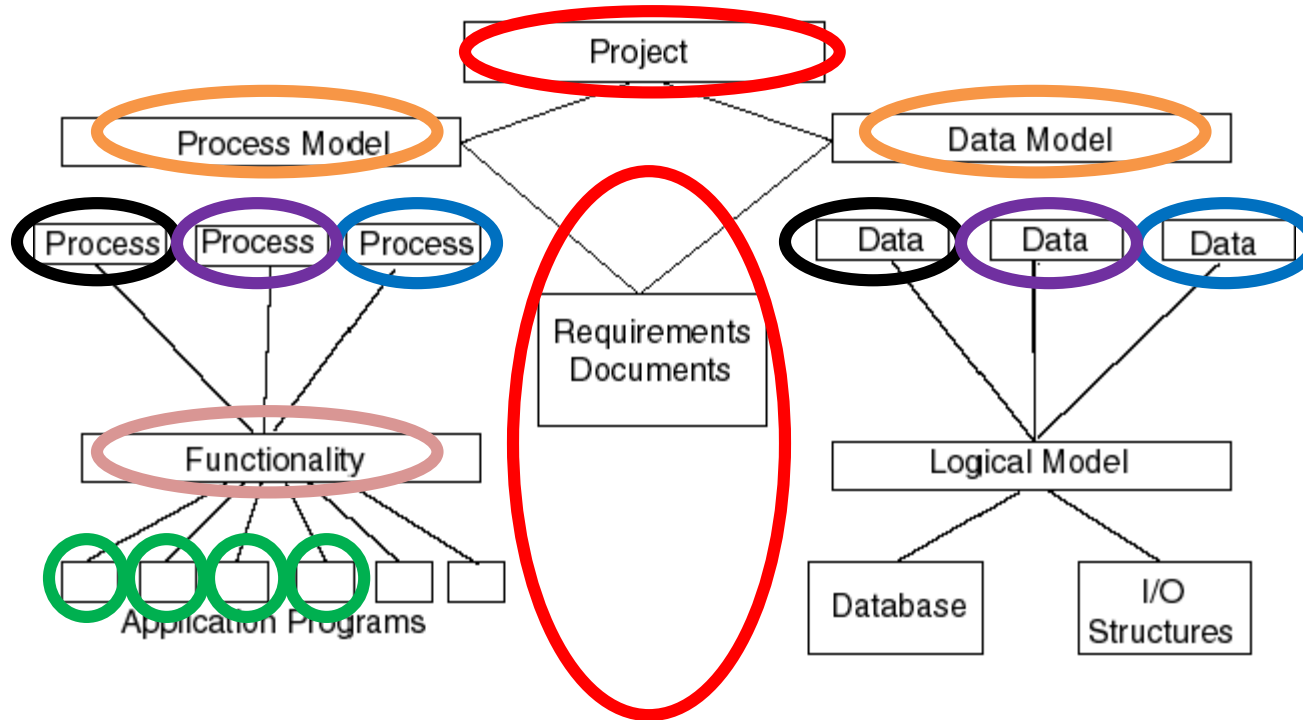
SDLC – Premise



SDLC – Premise



SDLC – Premise





• Generic Cohesion

- “Data” and “Process” exist – a correlation exists between them
- “Data” identifies and defines the information that will be utilized by the application to satisfy the business need
- “Process” defines how the “Data” will be utilized/leveraged to satisfy the business need

• Data

- **“Subject”** – highest level definition of user requirement - correlation with “System”
- **“Topics”** – breakdown of “Subject” - correlation with “Sub-Systems”
- **“Entities”** - breakdown of “Topics” - correlation with “Functions”

• Process

- **“System”** – highest level “process” – correlation with “Subject”
- **“Sub-System(s)”** – breakdown of “System” – correlation with “Topics”
- **“Function(s)”** – breakdown of “Sub-Systems” – correlation with “Entities”

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- **Software Development Lifecycle (SDLC) – Context and Premise**
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Leverage the Software Development Lifecycle

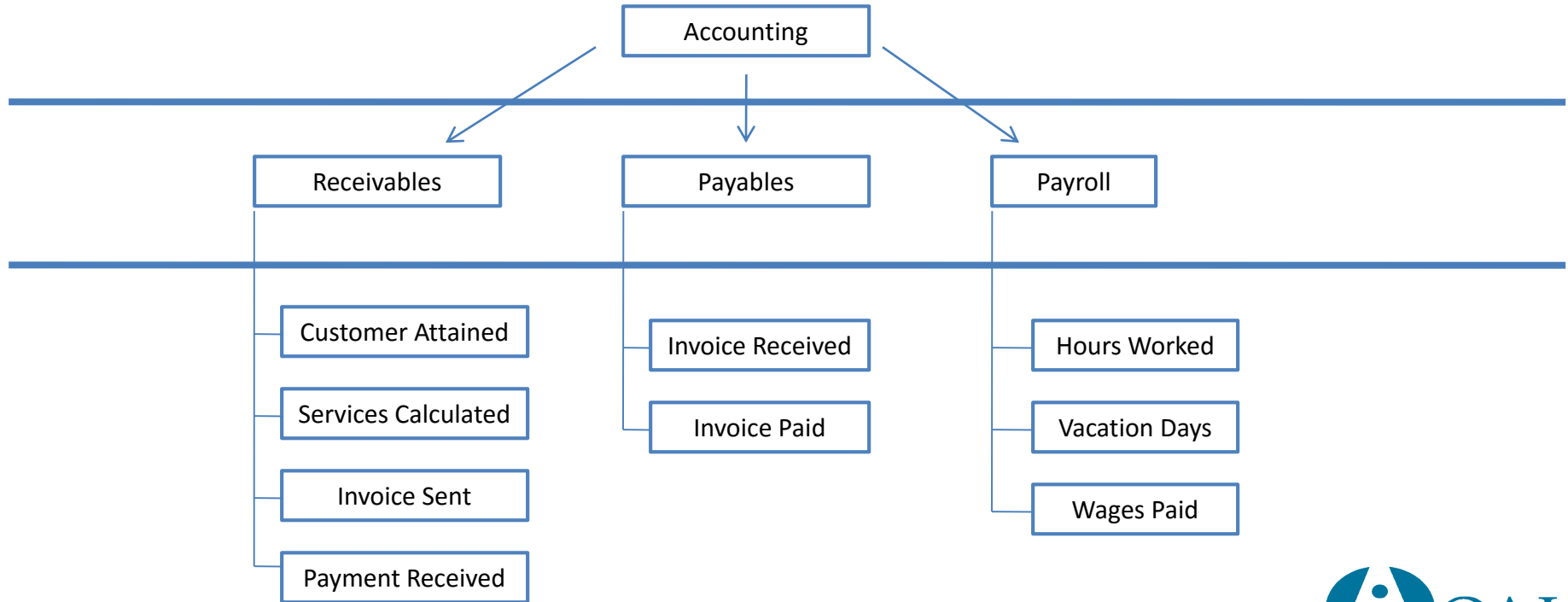
Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)**
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

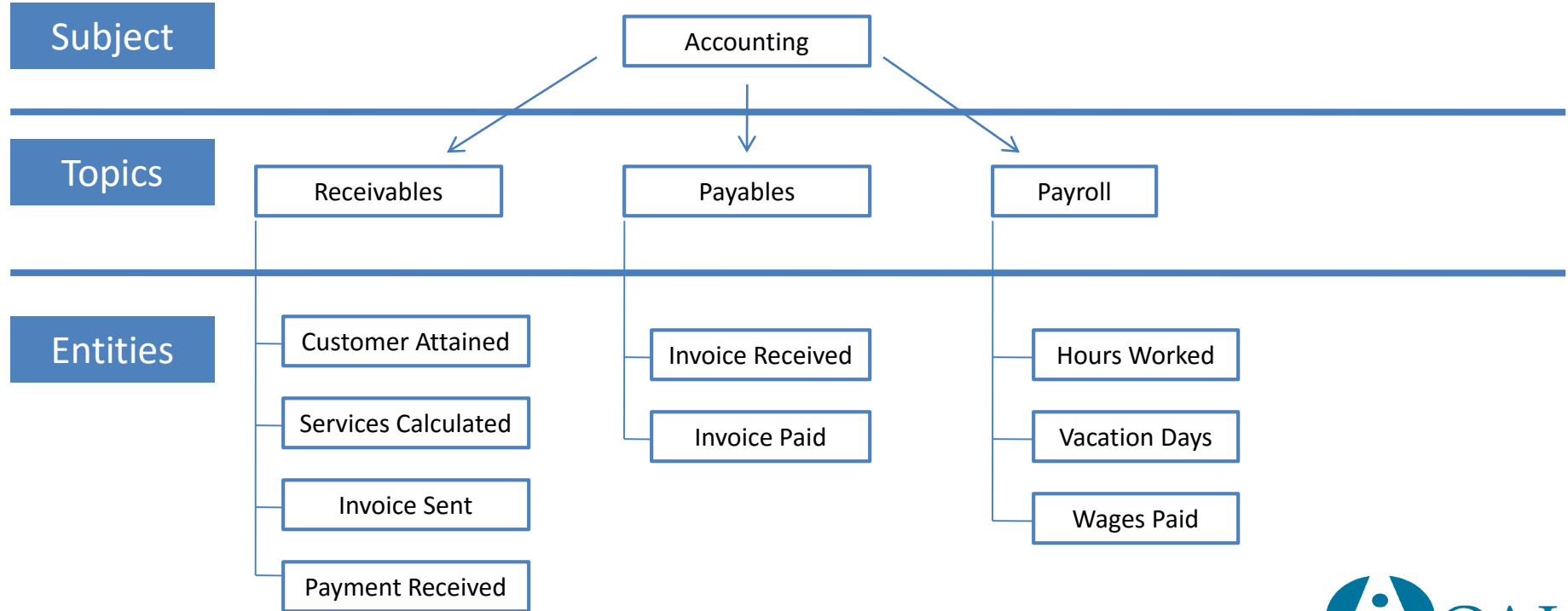
SDLC – Requirements (Identification and Definition)

Sample Application



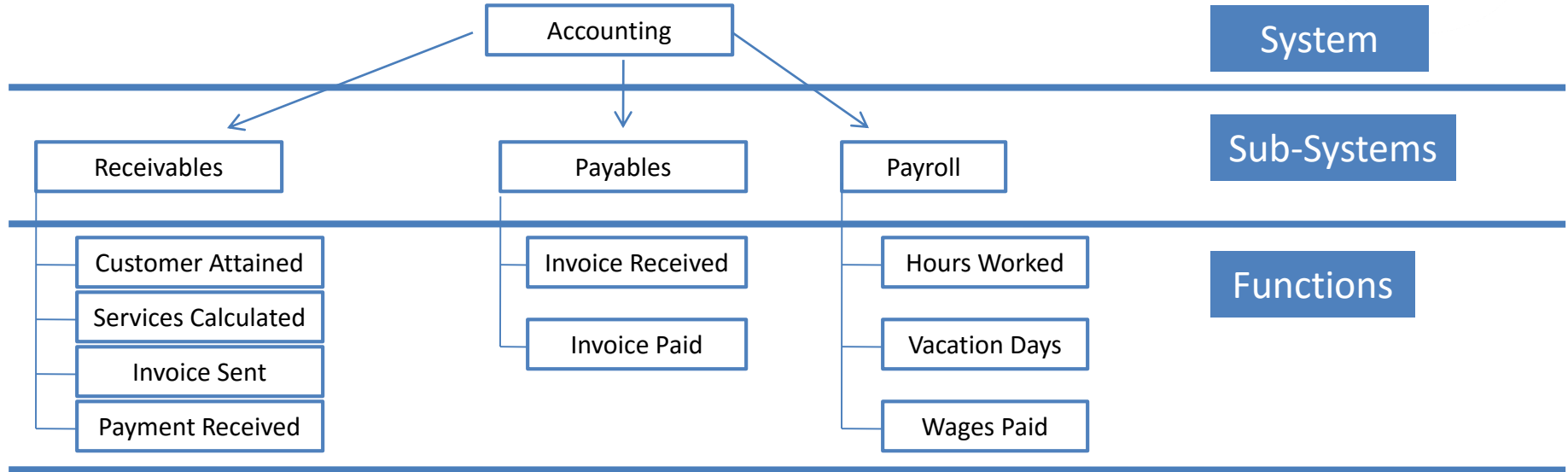
SDLC – Requirements (Identification & Definition)

“Data” Context



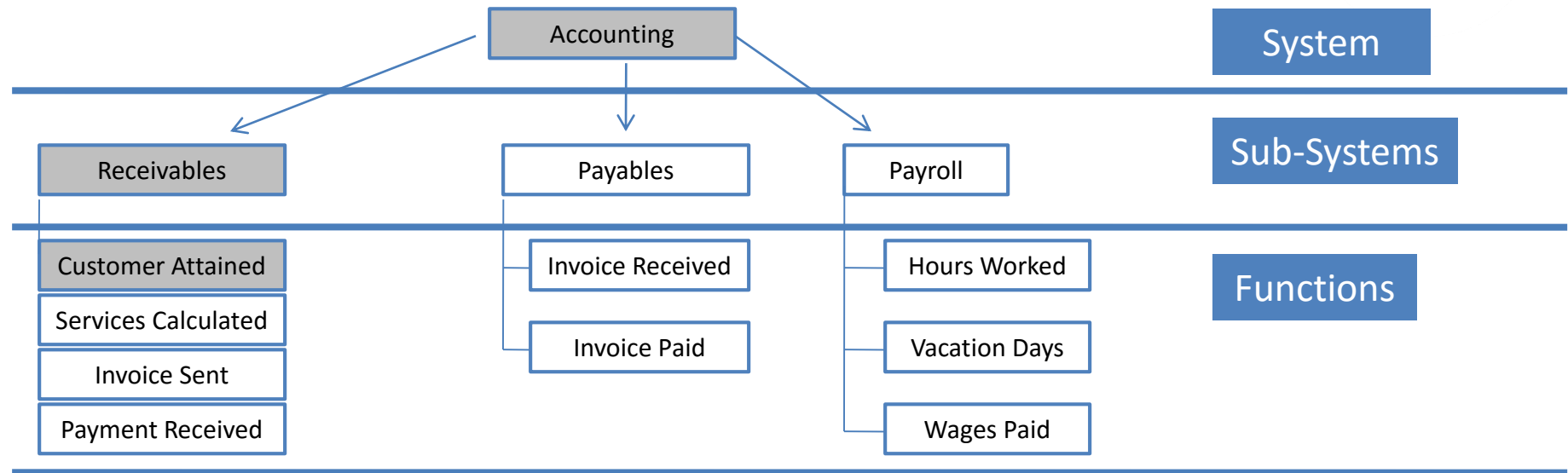
SDLC – Requirements (Identification & Definition)

“Process” Context



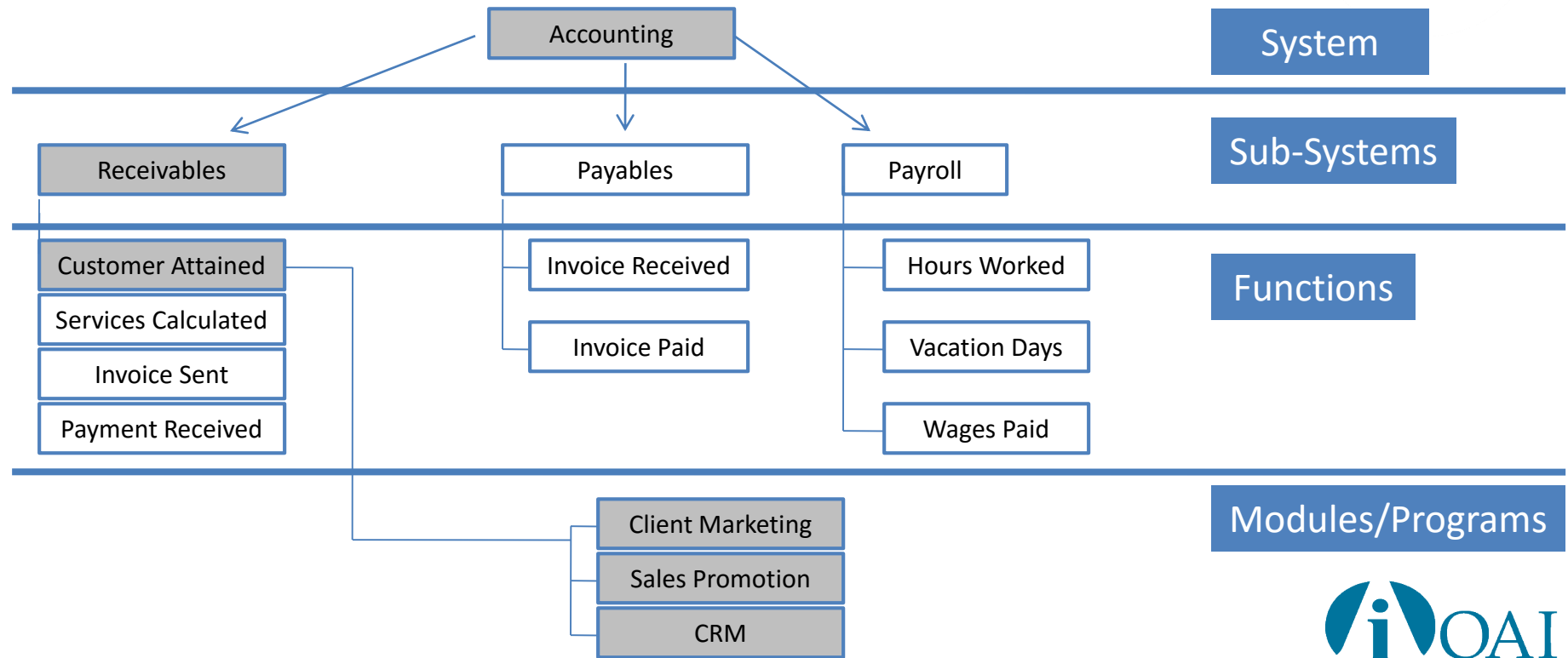
SDLC – Requirements (Identification & Definition)

“Process” Context



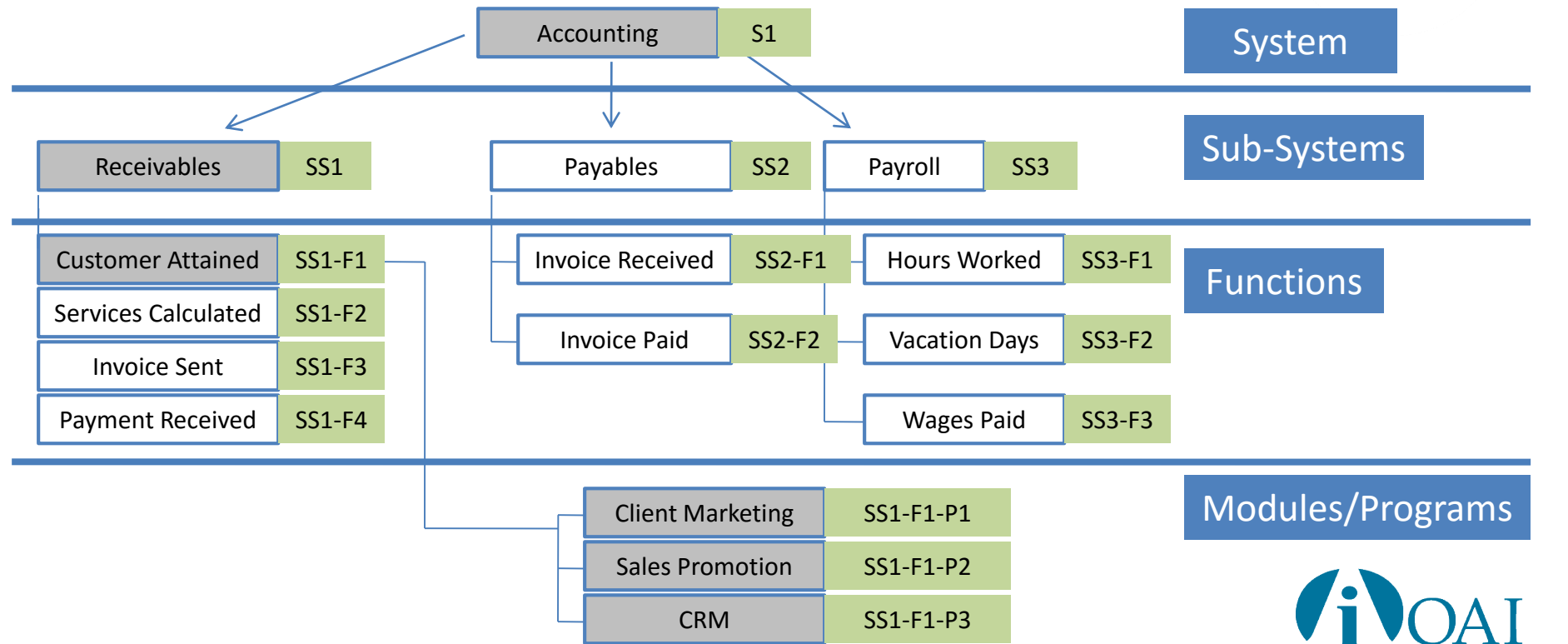
SDLC – Requirements (Identification & Definition)

“Process” Context



SDLC – Requirements (Identification & Definition)

“Process” – Naming Convention





• Generic Cohesion

- “Data” and “Process” exist – a correlation exists between them
- “Data” identifies and defines the information that will be utilized by the application to satisfy the business need
- “Process” defines how the “Data” will be utilized/leveraged to satisfy the business need

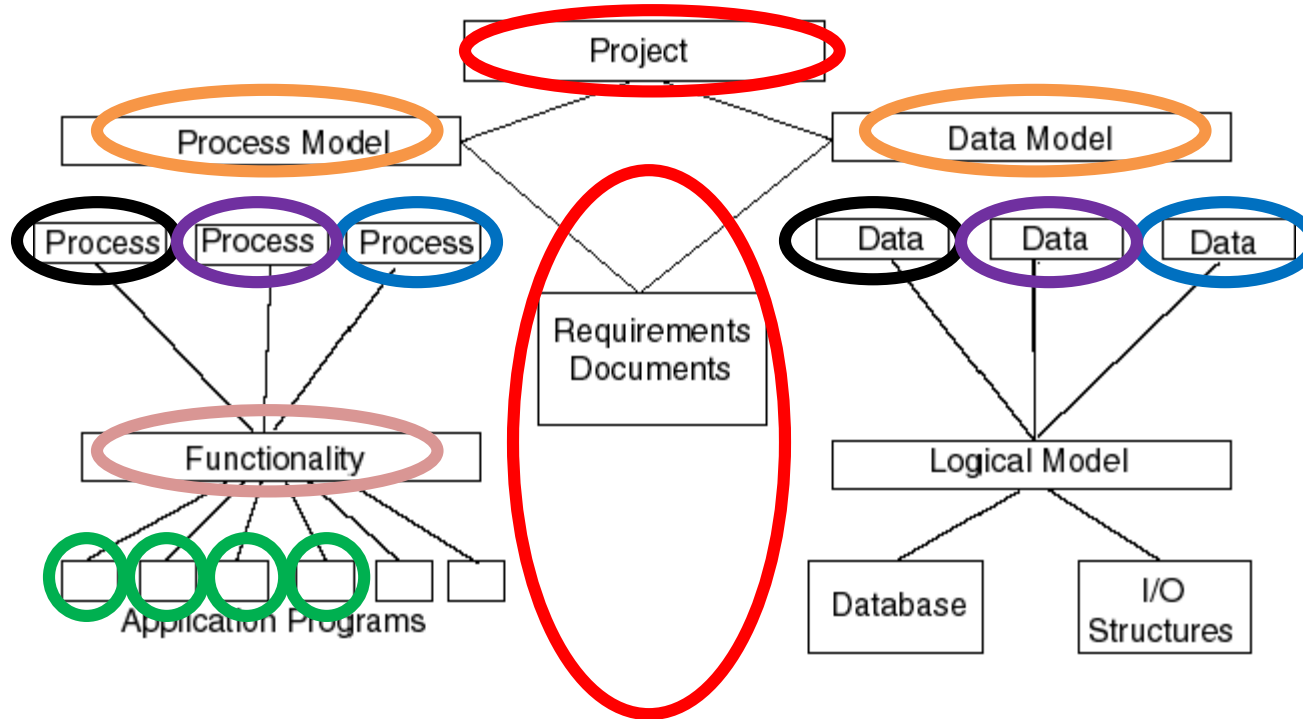
• Data

- **“Subject”** – highest level definition of user requirement - correlation with “System”
- **“Topics”** – breakdown of “Subject” - correlation with “Sub-Systems”
- **“Entities”** - breakdown of “Topics” - correlation with “Functions”

• Process

- **“System”** – highest level “process” – correlation with “Subject”
- **“Sub-System(s)”** – breakdown of “System” – correlation with “Topics”
- **“Function(s)”** – breakdown of “Sub-Systems” – correlation with “Entities”

SDLC – Premise



QAlassist Integrated Methodology



QAlassist Integrated Methodology

| | Initiate | Plan | Execute & Control | | Closeout |
|----------------------|---|---|--|---|--|
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Project Closeout Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| | Systems Analysis | Design | Build | Test | Release |
| Software Development | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

QAlassist Integrated Methodology

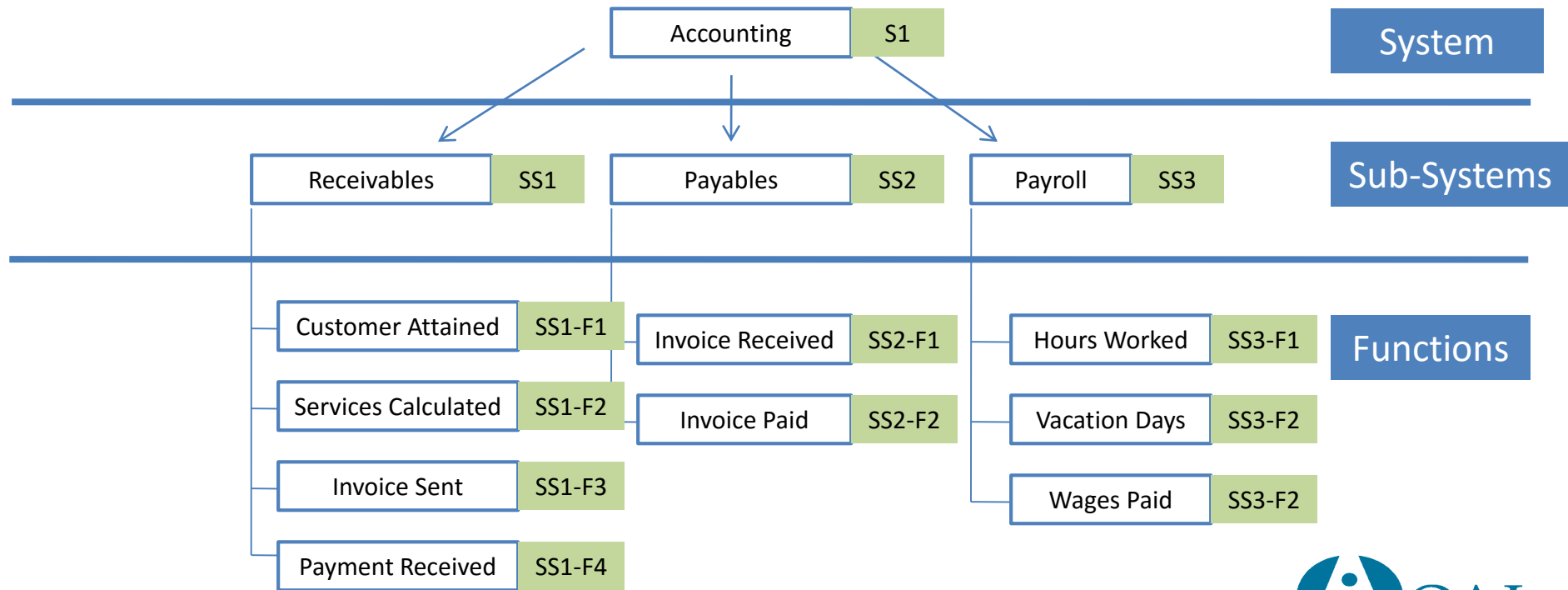


QAlassist Integrated Methodology

| | Initiate | Plan | Execute & Control | | Closeout |
|----------------------|---|--|---|---|--|
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/WBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization Project Closeout |
| Software Development | Systems Analysis | Design | Build | Test | Release |
| | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Tracability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Tracability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

SDLC - Mechanics

High Level Solution Design Deliverable (S1, SSX, SSX-FX)



QAlassist Integrated Methodology

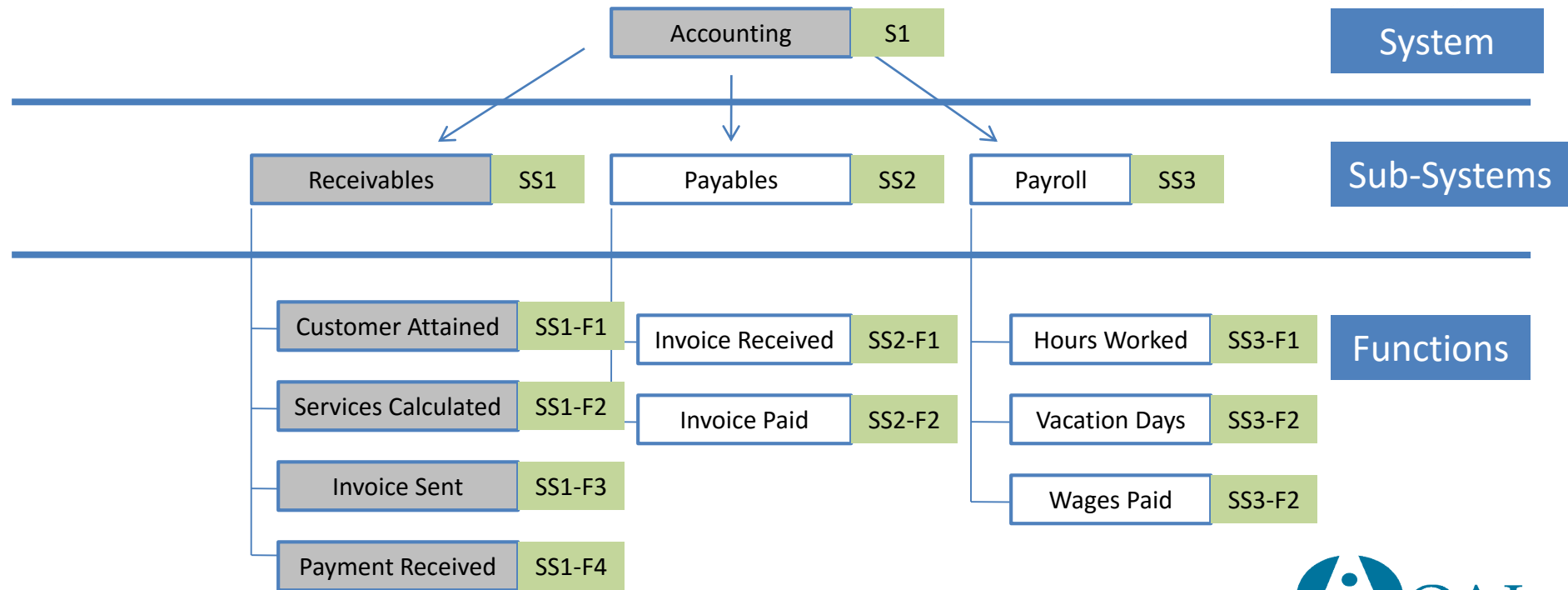


QAlassist Integrated Methodology

| | Initiate | Plan | Execute & Control | | Closeout |
|----------------------|---|---|--|---|--|
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Project Closeout |
| Software Development | Systems Analysis | Design | Build | Test | Release |
| | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

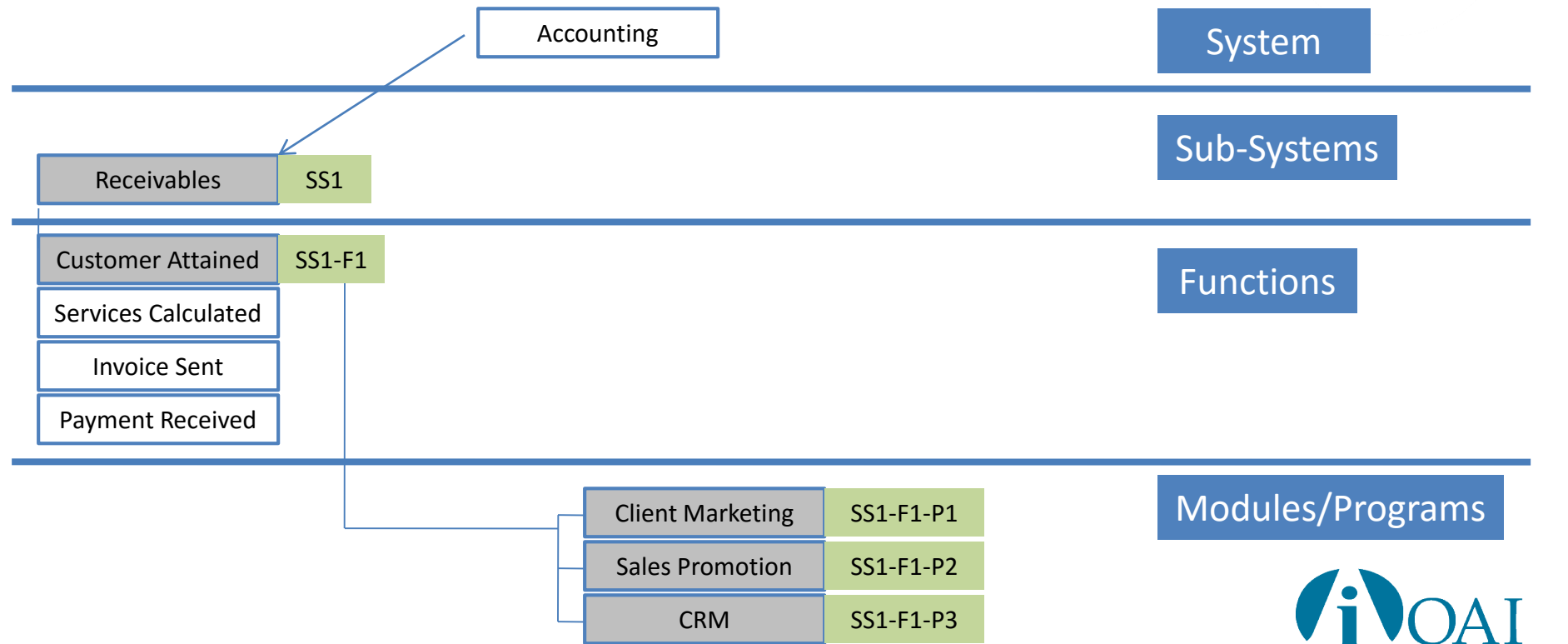
SDLC - Mechanics

Detail Solution Design Deliverables (SSX-FX one per)



SDLC - Mechanics

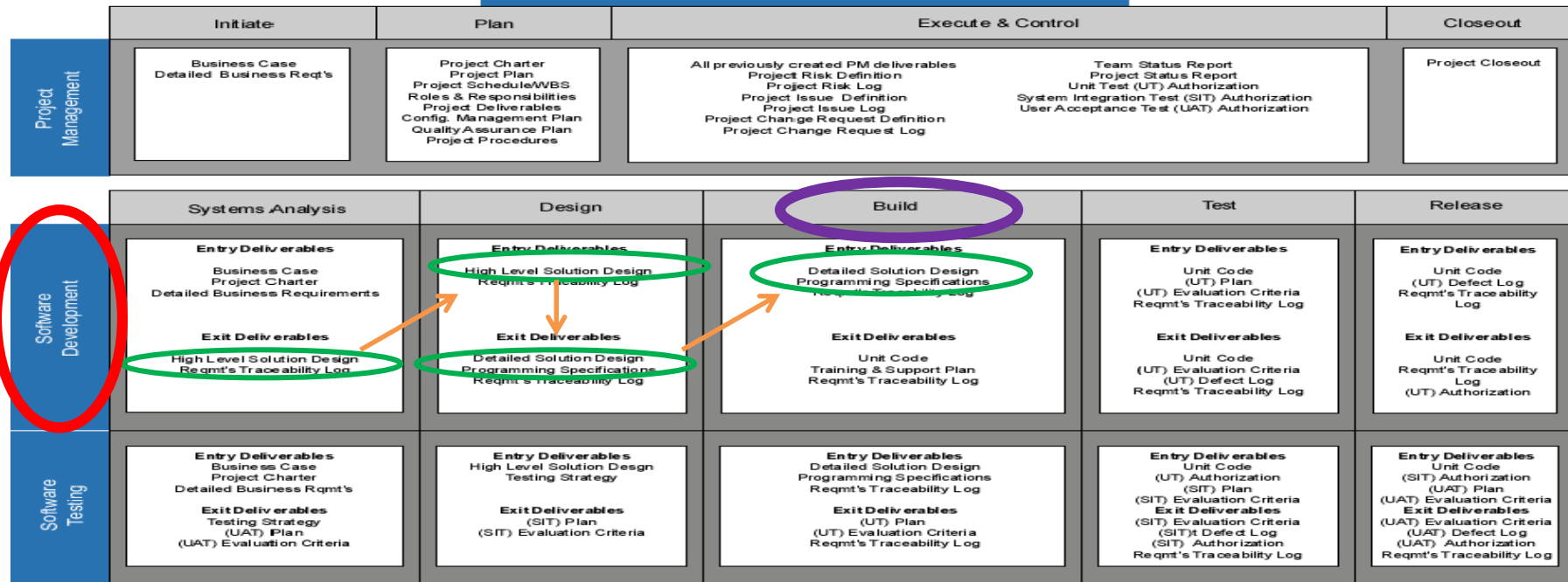
Programming Specifications (SSX-FX-PX one per)



QAlassist Integrated Methodology



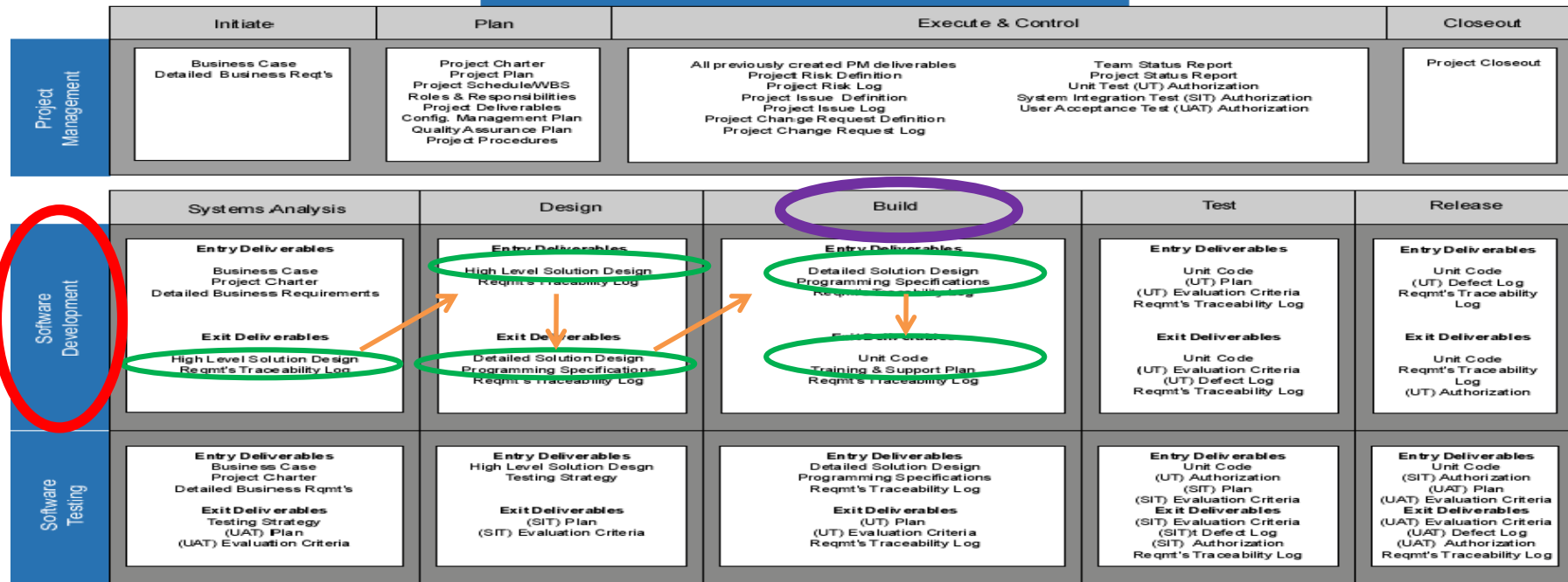
QAlassist Integrated Methodology

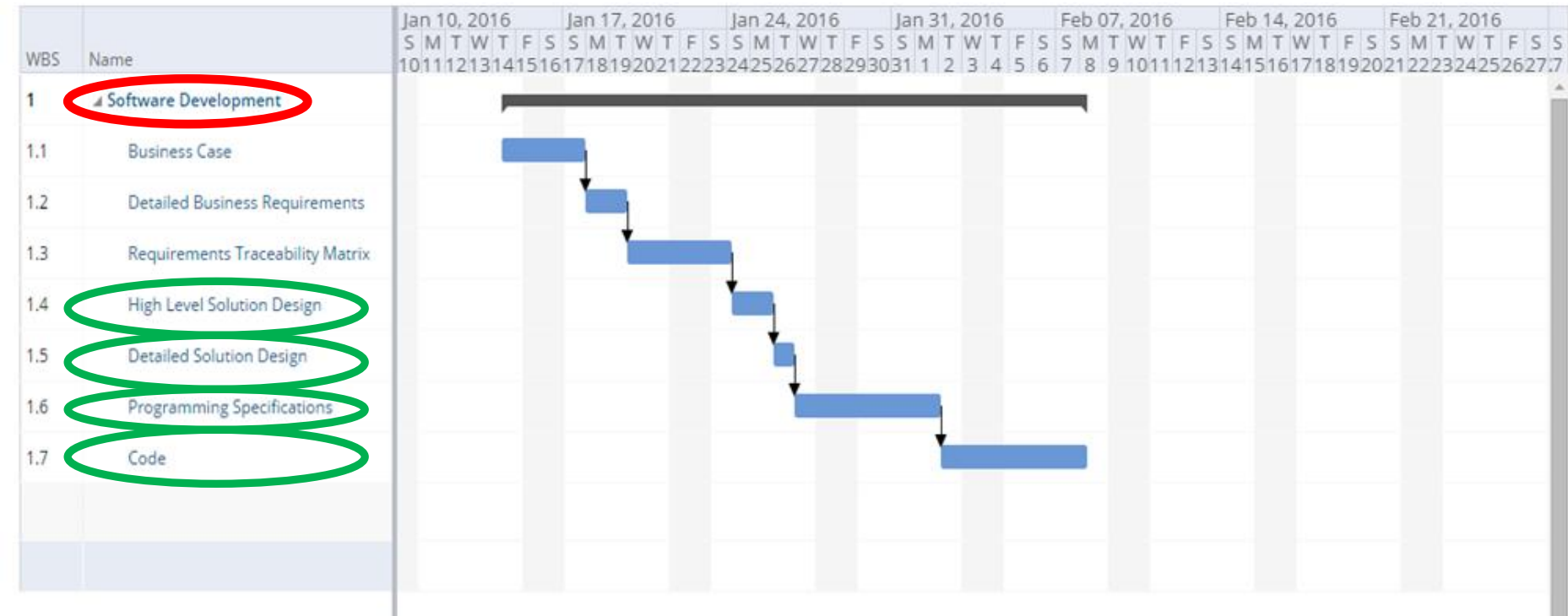


QAlassist Integrated Methodology



QAlassist Integrated Methodology





Software Development Requirements Deliverables & Dependencies



► High Level Solution Design deliverable

- Identifies data “Subject” and process “Systems”
- Identifies data “Topics” and process “Subsystems”
- Identifies data “Entities” and process “Functions”

► Detailed Solution Design deliverable(s)

- One deliverable for every “Function” as identified in the **High Level Solution Design** deliverable
- Functionality defined (Manual and Automated)
- All required “Unit Modules/Programs” are identified

► Programming Specification deliverable(s)

- One deliverable for every “Unit Module/Program” as identified in the **Detailed Solution Design** deliverables
- Specifics for each “Unit Module/Program” are described

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- **SDLC - Requirements (Identification and Definition)**
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- **SDLC Phases - Objectives, Iterations, Traceability**
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concept” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- **SDLC Phases - Objectives, Iterations, Traceability**
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Software Development Lifecycle (SDLC) Objectives



- Establish a foundation for ongoing collaboration and communication between Business/User Community and Information Technology/IT resources
- Business/User community contributes to and authorizes formal requirements and deliverables throughout the completion of the project – necessary documentation
- Computer **Unit Code** is developed/built to reflect the authorized and required functionality (design and business requirements)
- Computer **Unit Code** is successfully **unit tested** against the authorized design and business requirements - satisfactory code is made available for additional testing
- End User training is planned and delivered



● SDLC objectives (see previous page) are achieved utilizing five phases, they are :

- Systems Analysis
- Design
- Build
- Test (Unit)
- Release

IT Methodology

SDLC



QAassist Integrated Methodology

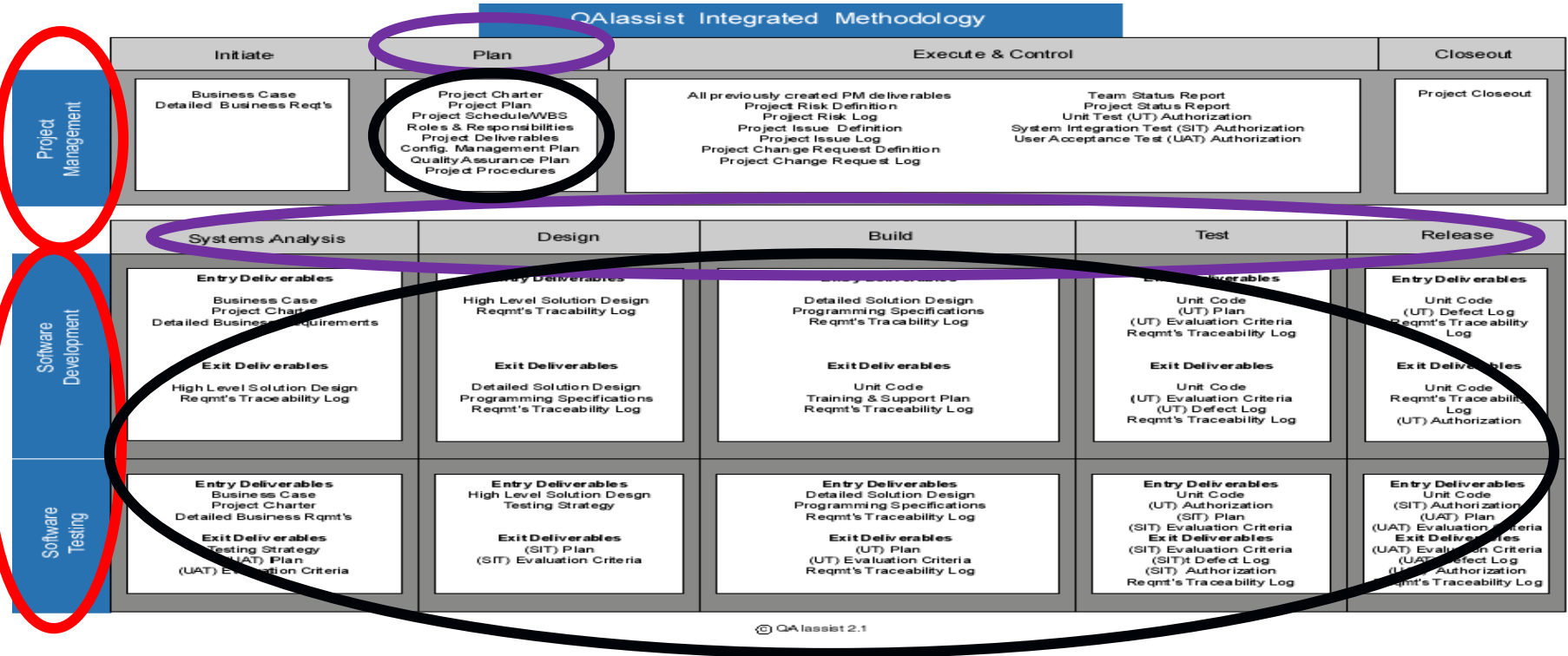
| | QAassist Integrated Methodology | | | | |
|----------------------|---|--|---|---|--|
| | Initiate | Plan | Execute & Control | | Closeout |
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| Software Development | Systems Analysis | Design | Build | Test | Release |
| | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

IT Methodology

SDLC

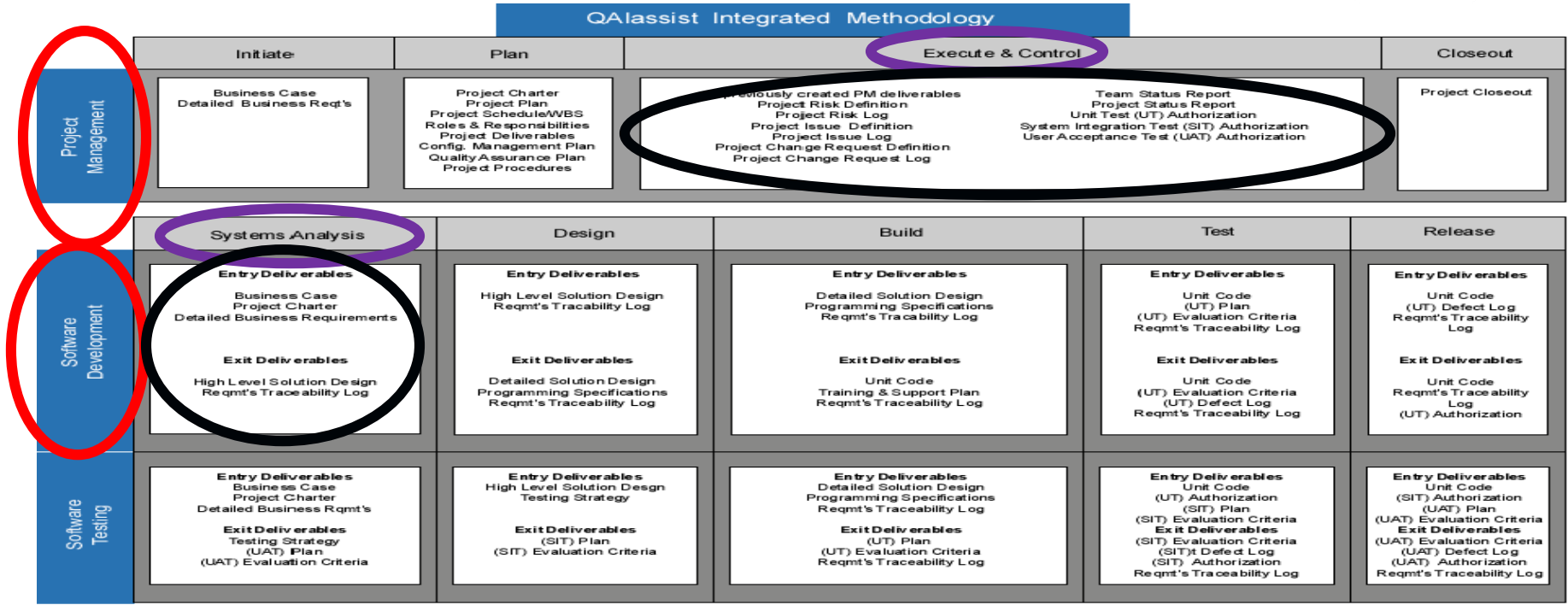


QAassist Integrated Methodology



IT Methodology

SDLC – Systems Analysis Phase



SDLC – Systems Analysis Phase



► Objectives

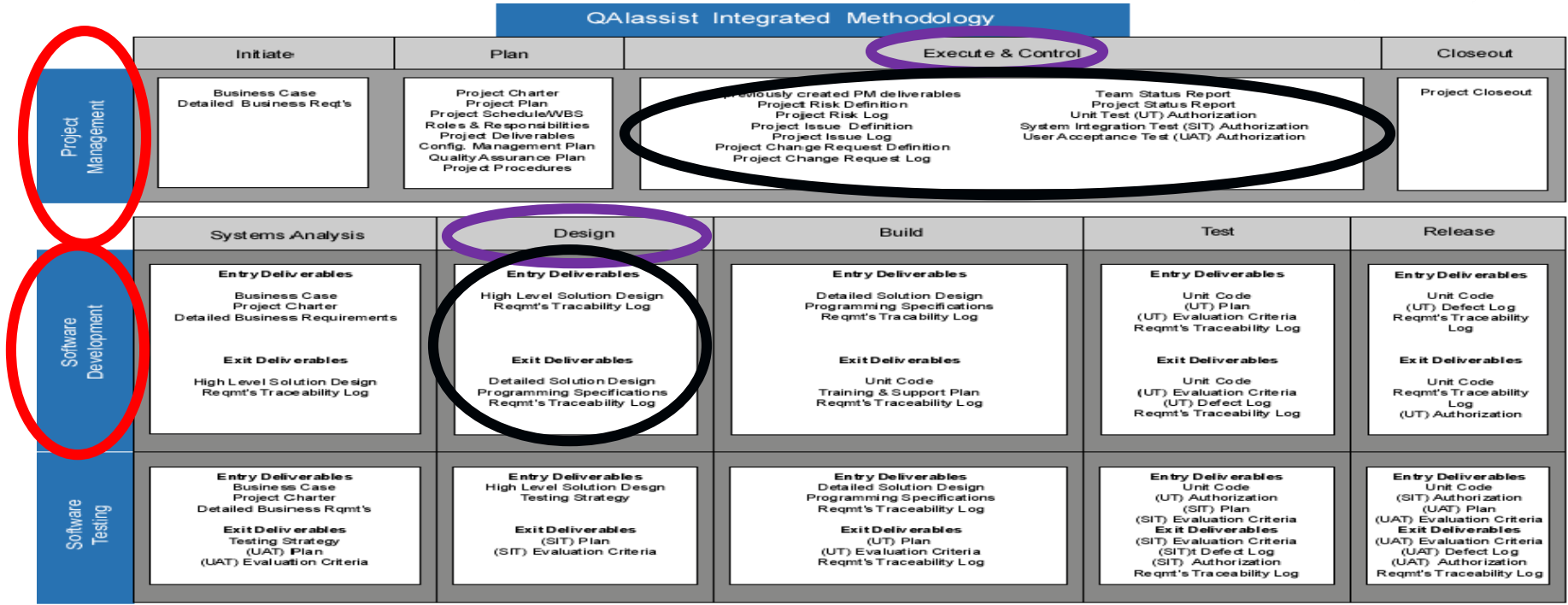
- ensure a formal project exists and has been authorized (**Business Case** established)
- project team members review and understand project context (deliverables)
- identify/refine initial business requirements
- establish (**High Level Solution Design**)
- establish (**Requirements Traceability Log**)

► Iterations

- iterative within the phase (waterfall)
- iterative within the lifecycle (agile)

IT Methodology

SDLC – Design Phase



SDLC – Design Phase



► Objectives

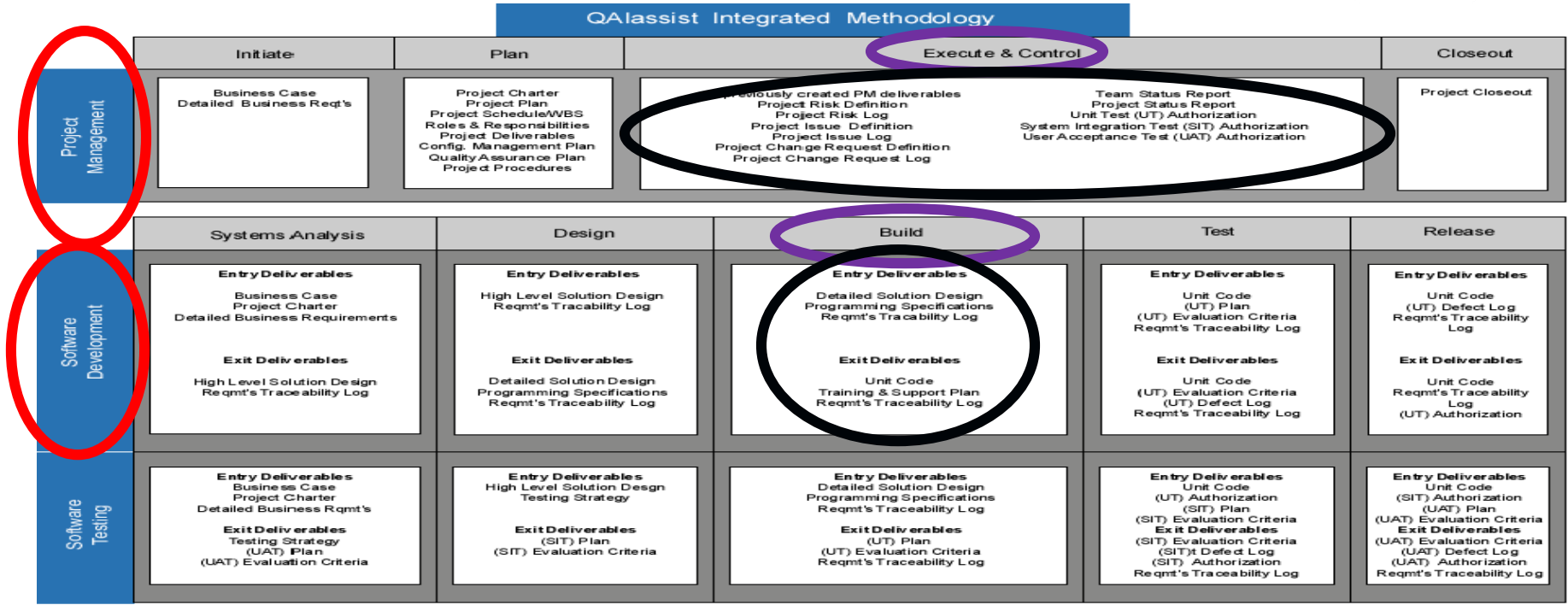
- ensure the application is designed in accordance with the authorized requirements defined during the System Analysis phase – (**High Level Solution Design**)
- complete the design of the application – (**Detailed Solution Design**)
- create specifications that reflect the authorized design & business requirements - (**Programming Specification (s)**)
- maintain (**Requirements Traceability Log**)

► Iterations

- iterative within the phase (waterfall)
- iterative within the lifecycle (agile)

IT Methodology

SDLC – Build Phase



SDLC – Build Phase



► Objectives

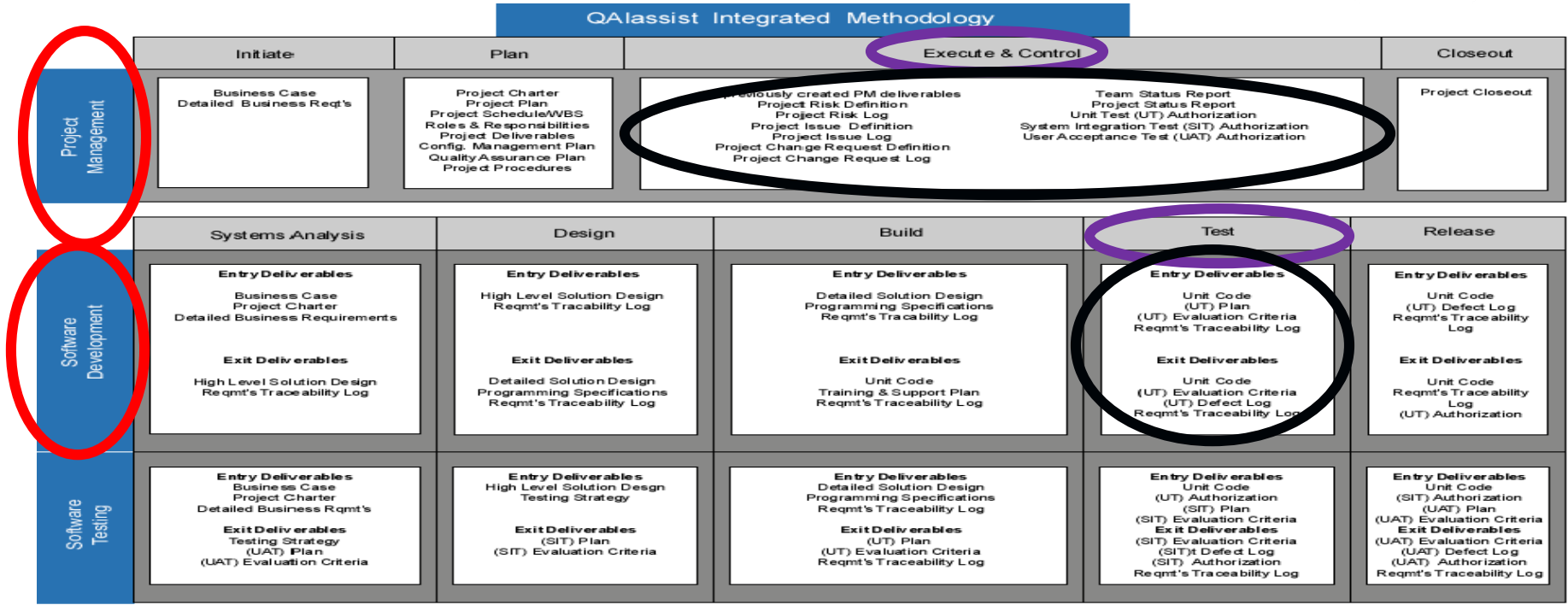
- develop the required functionality according to the authorized design and business requirements
(High Level Solution Design, Detailed Solution Design, Programming Specifications)
- create **(Unit Code)** of the application according to the programming specifications
- develop the **(User Training and Support Plan)**
- maintain **(Requirements Traceability Log)**

► Iterations

- iterative within the phase (waterfall)
- iterative within the lifecycle (agile)

IT Methodology

SDLC – Test Phase



SDLC – Test Phase



► Objectives

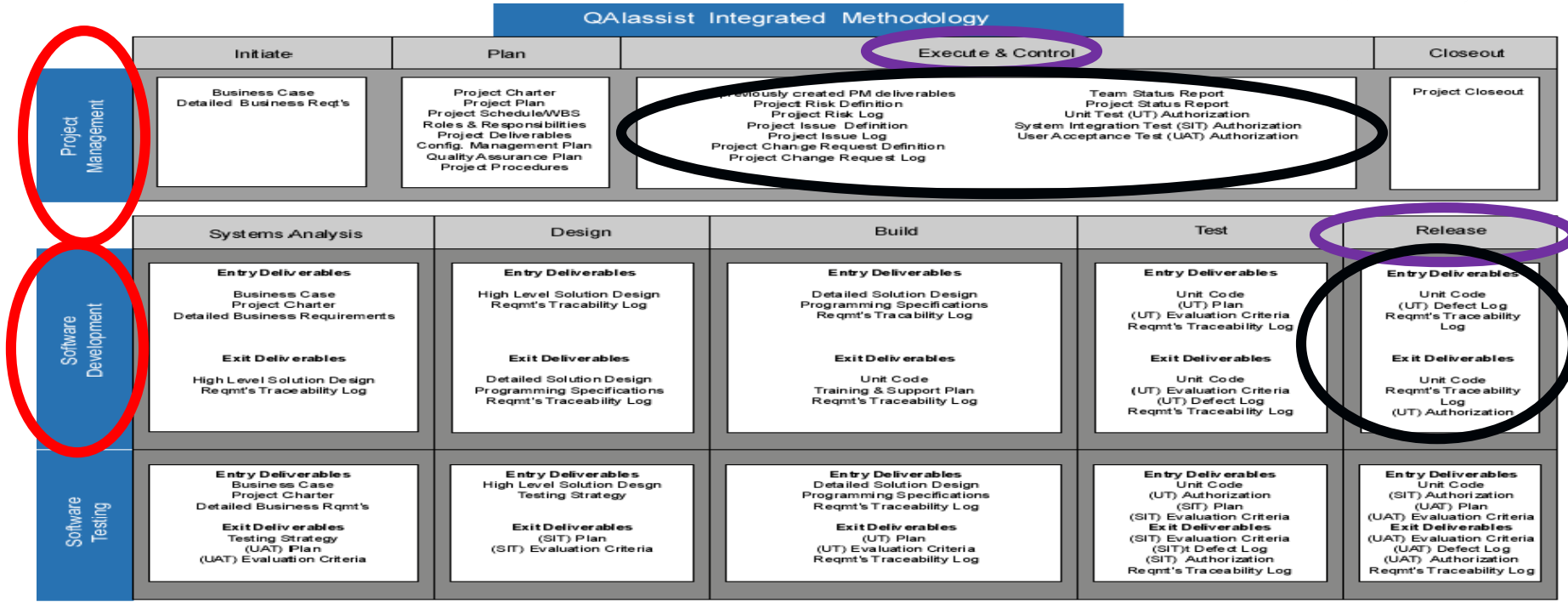
- perform all unit testing according to (**Unit Test Plans** - from STLC)
- Identify, log (**Unit Test Defect Log**) and address all conditions that do not satisfy unit testing criteria
- ensure the application/product has been built in accordance with the authorized design, business requirements, (**Unit Test Evaluation Criteria** - from STLC)
- ensure all unit testing deliverables/work products are under configuration management
- maintain **Requirements Traceability Log**

► Iterations

- iterative within the phase (waterfall)
- iterative within the lifecycle (agile)

IT Methodology

SDLC – Release Phase



SDLC – Release Phase



► Objectives

- authorize that all necessary “unit” testing is complete - promote the newly built and unit tested functionality/application for additional testing (**Unit Test Authorization**)
- ensure all unit testing deliverables and work products under configuration management
- maintain **Requirements Traceability Log**
- migrate the application for additional (SIT, UAT) testing

► Iterations

- iterative within the phase (waterfall)
- iterative within the lifecycle (agile)

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- **SDLC Phases - Objectives, Iterations, Traceability**
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- **SDLC Phases** - Objectives, Iterations, **Traceability**
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

IT Methodology

SDLC



QAassist Integrated Methodology

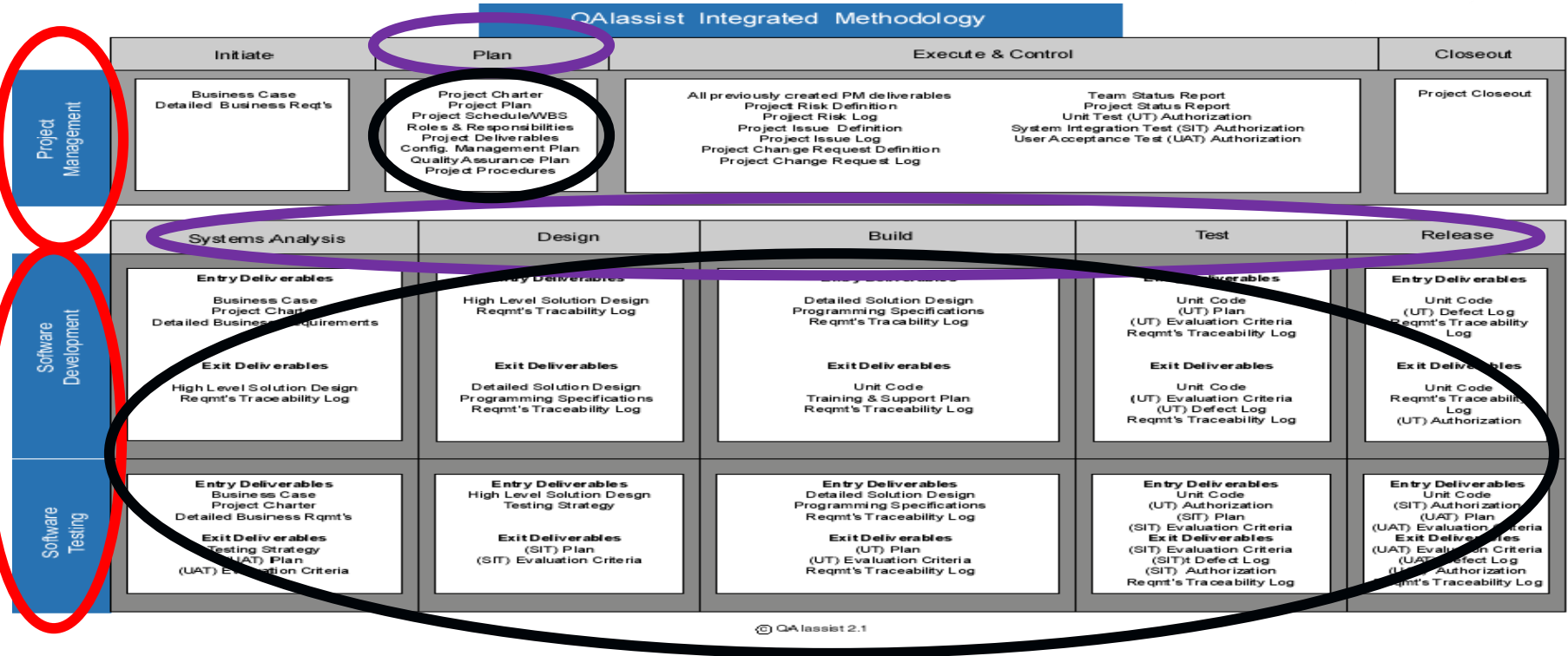
| | QAassist Integrated Methodology | | | | |
|----------------------|---|--|---|---|--|
| | Initiate | Plan | Execute & Control | | Closeout |
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| Software Development | Systems Analysis | Design | Build | Test | Release |
| | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

IT Methodology

SDLC

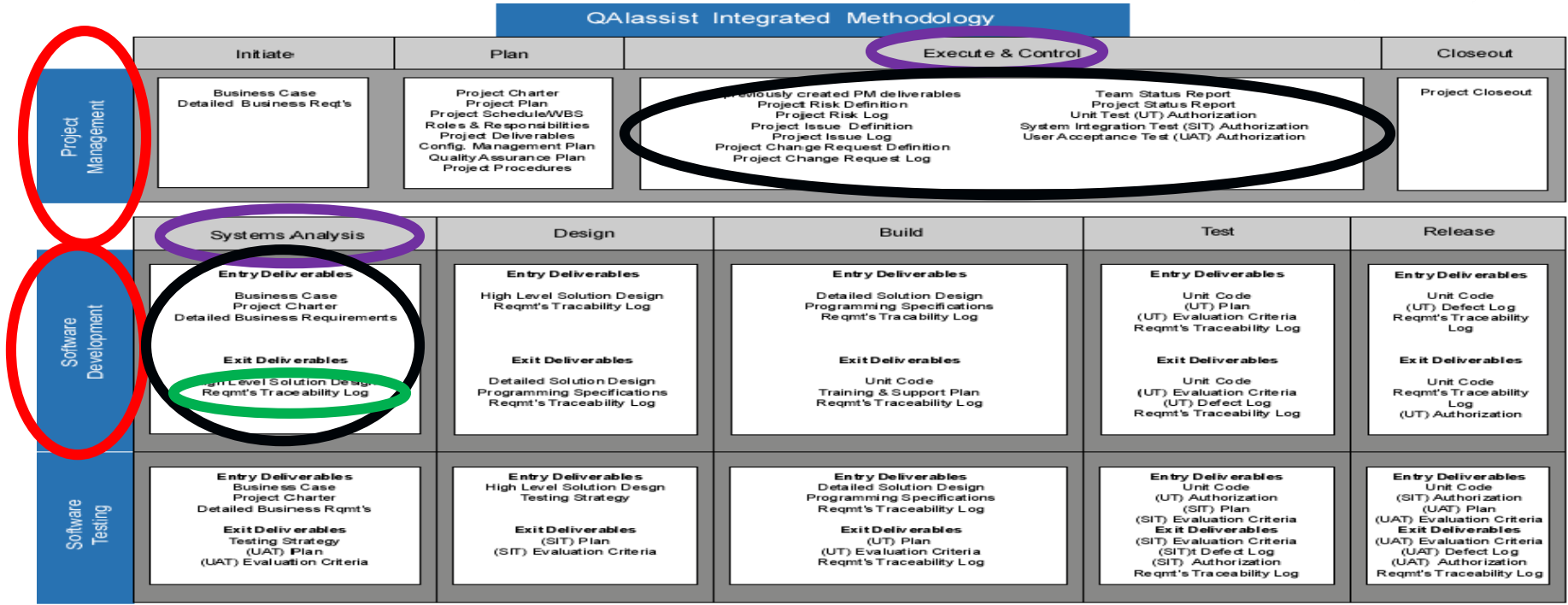


QAassist Integrated Methodology



IT Methodology

SDLC – Requirements Traceability



IT Methodology

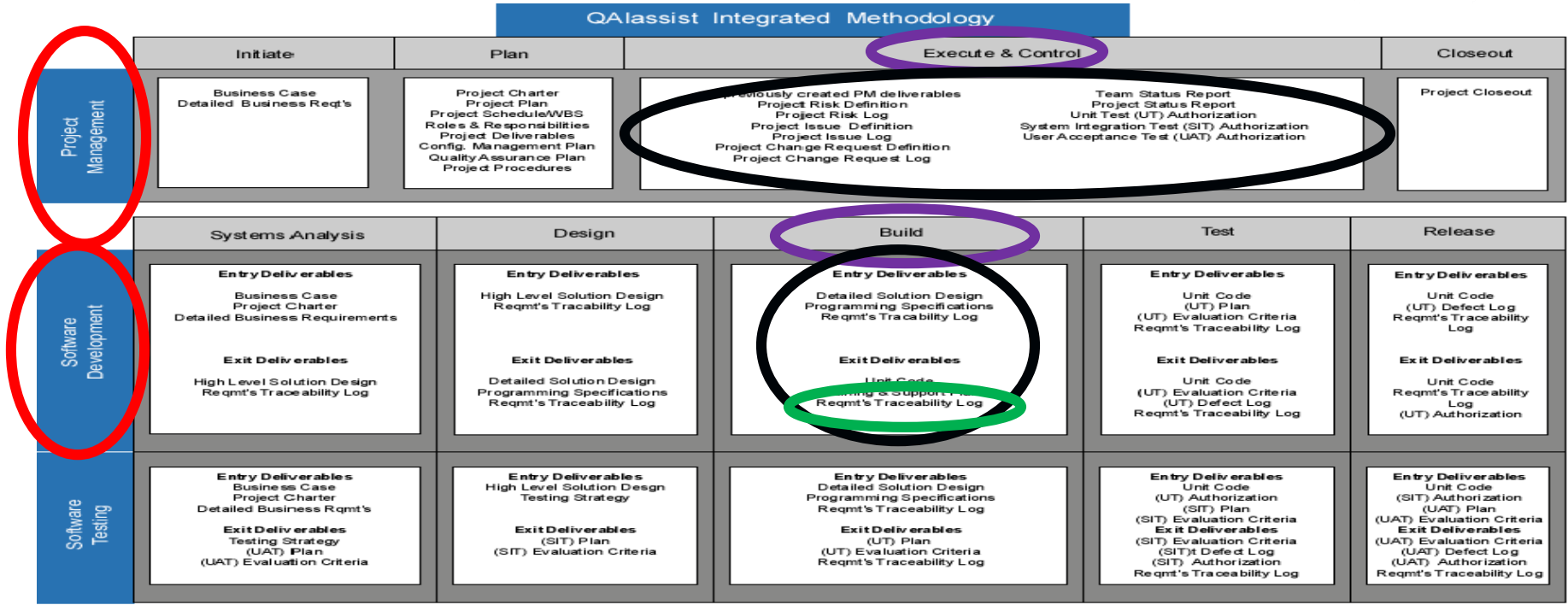
SDLC – Requirements Traceability



| QAassist Integrated Methodology | | | | | |
|---------------------------------|---|---|--|---|--|
| | Initiate | Plan | Execute & Control | Closeout | |
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/WBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | Previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization | |
| | | | | | |
| | Systems Analysis | Design | Build | Test | Release |
| Software Development | Entry Deliverables Business Case Project Charter Detailed Business Requirements Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Entry Deliverables High Level Solution Design Reqmt's Traceability Log Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Defect Log Reqmt's Traceability Log Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables Business Case Project Charter Detailed Business Reqmt's Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Entry Deliverables High Level Solution Design Testing Strategy Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Entry Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Entry Deliverables Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Entry Deliverables Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

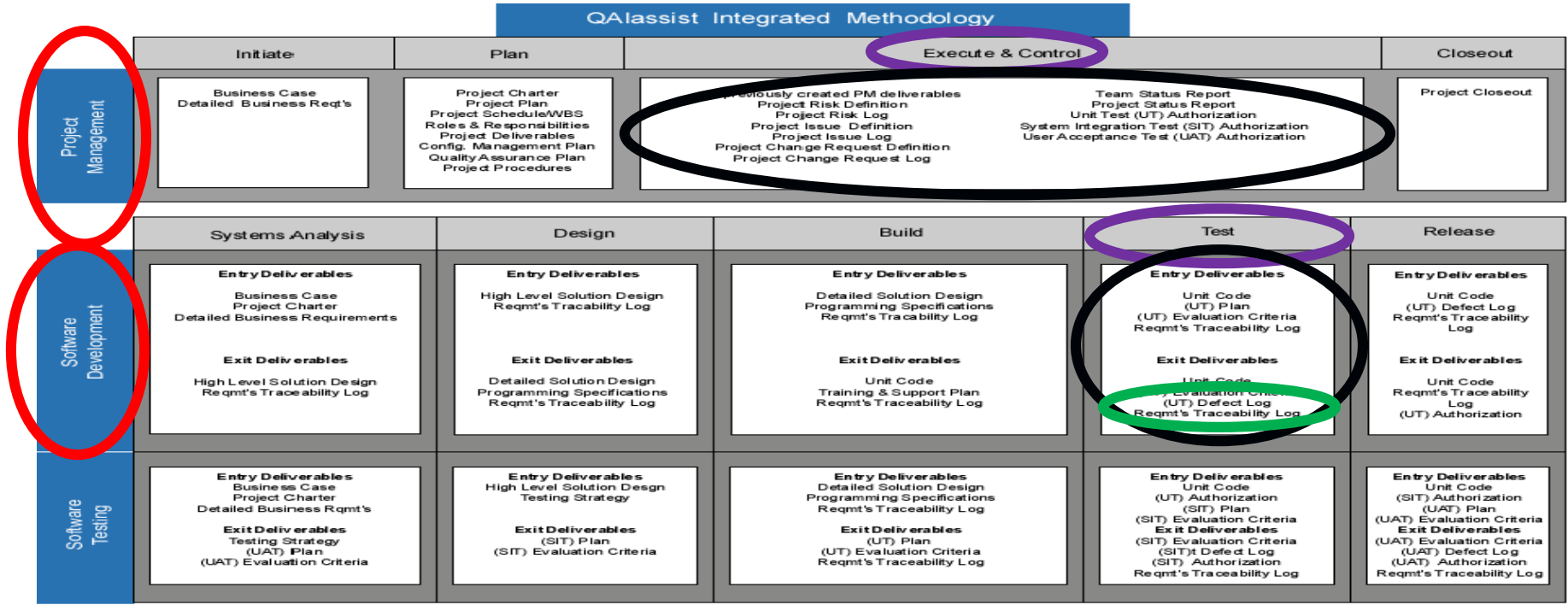
IT Methodology

SDLC – Requirements Traceability



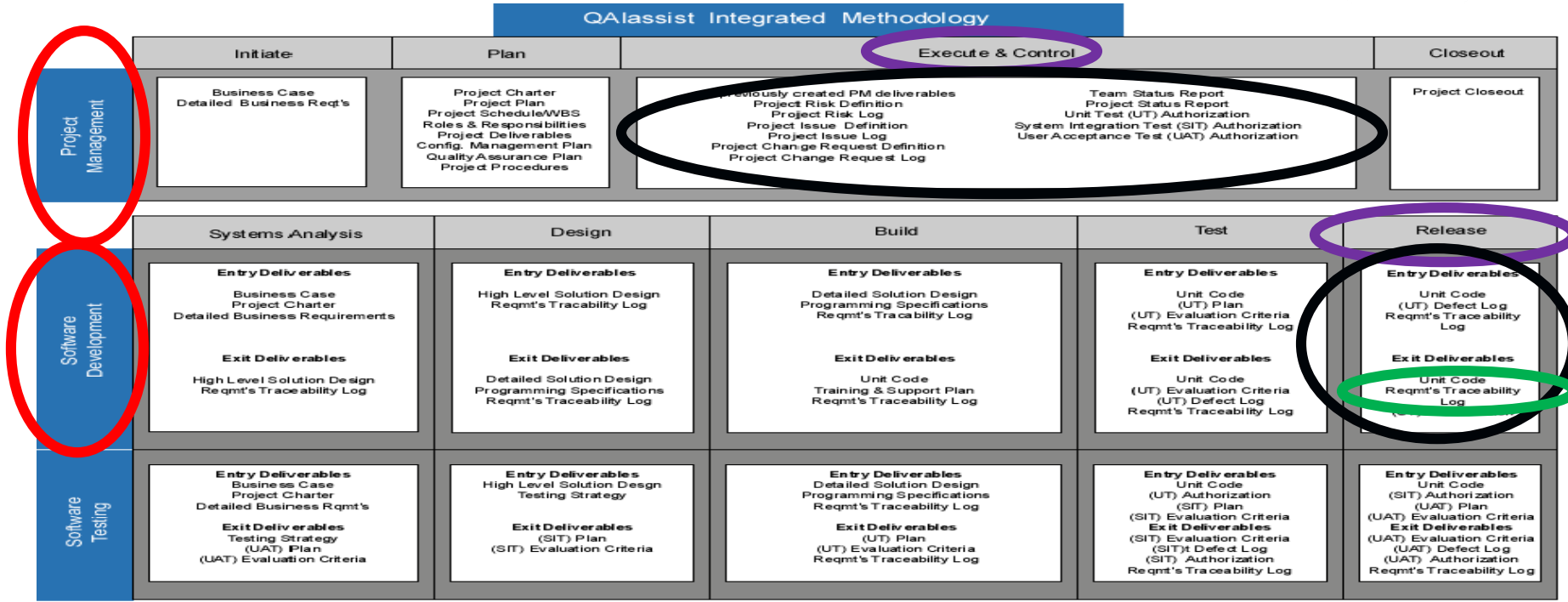
IT Methodology

SDLC – Requirements Traceability



IT Methodology

SDLC – Requirements Traceability



Software Development Lifecycle (SDLC)

Requirements Traceability



- User Requirements are dynamic – they are defined, clarified and authorized as project evolves
- User Requirements are calibrated (large to small) – calibration reflected in deliverables
- User Requirements are maintained throughout life of project – documented accordingly
- User Requirements created/established (by SDLC Phase)
 - Systems Analysis (Detailed Requirements, High Level Solution Design)
 - Design (Detailed Solution Design)
 - Build (Programming Specifications)
- User Requirements may be revised/altered/appended to during
 - Unit Testing
 - Integration Testing
 - User Acceptance Testing

Requirements Traceability Log (Matrix) Reference/Access/Updating



► SDLC

- Systems Analysis Phase (Detailed Requirements, High Level Solution Design)
- Design Phase (Detailed Solution Design)
- Build Phase (Programming Specification)
- Test Phase (Unit Test Defect Log) ***

Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- **SDLC Phases** - Objectives, Iterations, **Traceability**
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- Recap

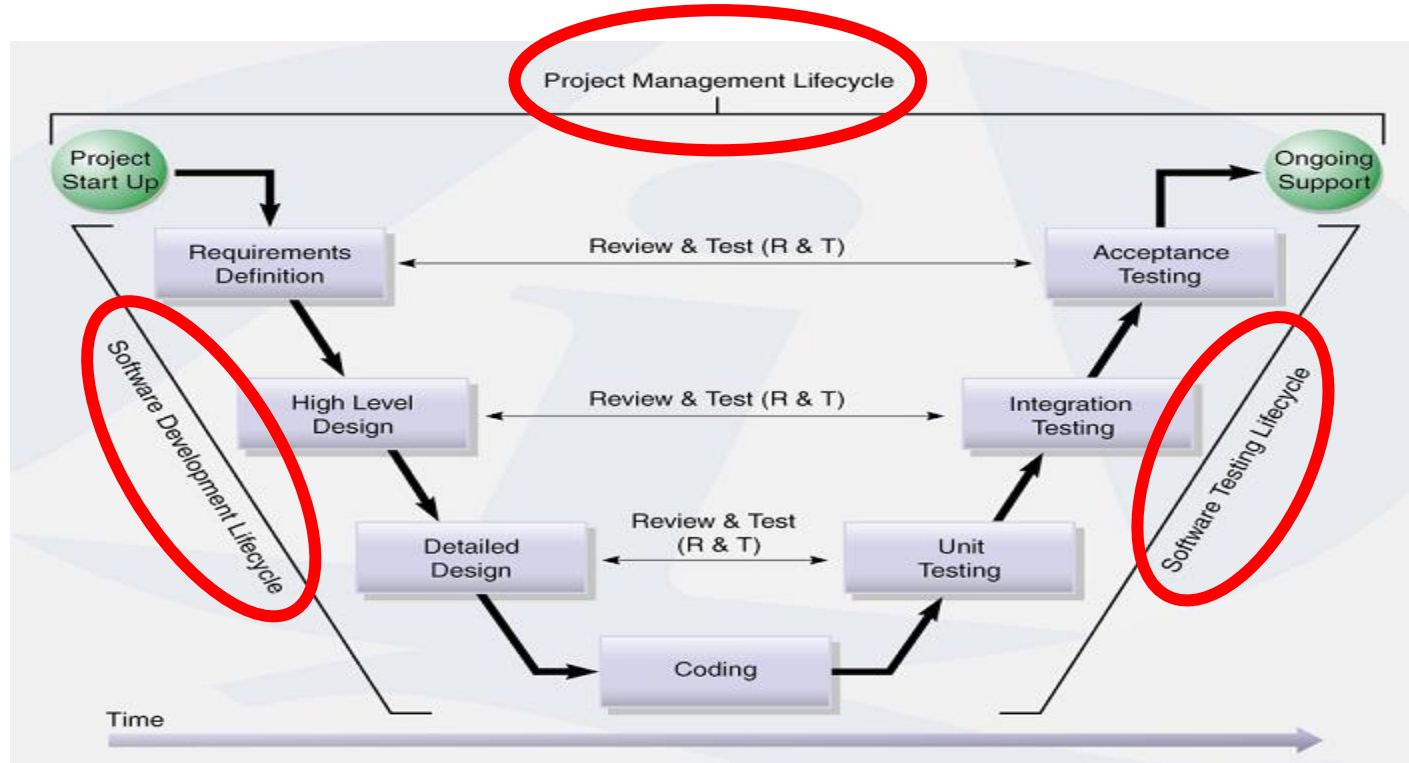
Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- **SDLC – Internal Dynamics (PMLC and STLC interfaces)**
- Recap

SDLC Integration



Internal Dynamics - SDLC Interfaces



- Project Management Lifecycle

- Initial Project Planning (Project Plan and Project Schedule)
- Ongoing Project Execution (Issue Log, Risk Log, Change Request, Status)

- Software Testing Lifecycle

- Unit Test Planning, Execution & Authorization

Internal Dynamics - SDLC Interfaces



► Project Management Lifecycle

- Initial Project Planning (Project Plan and Project Schedule)
- Ongoing Project Execution (Issue Log, Risk Log, Change Request, Status)

► Software Testing Lifecycle

- Unit Test Planning, Execution & Authorization

Project Management with SDLC

Initial Project Planning

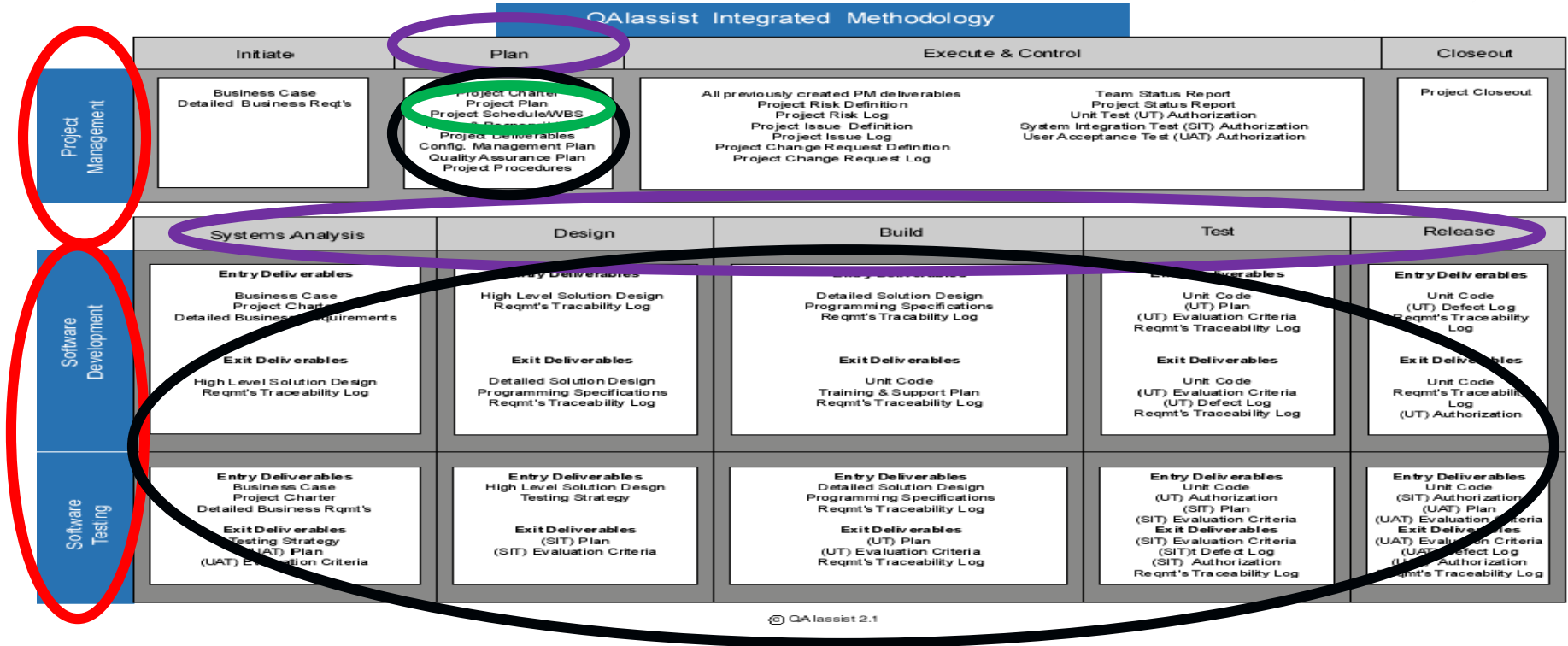


QAassist Integrated Methodology

| | QAassist Integrated Methodology | | | | |
|----------------------|--|--|--|--|---|
| | Initiate | Plan | Execute & Control | | Closeout |
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| | Systems Analysis | Design | Build | Test | Release |
| | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables |
| Software Development | Business Case Project Charter Detailed Business Requirements | High Level Solution Design Reqmt's Traceability Log | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Unit Code (UT) Defect Log Reqmt's Traceability Log |
| | Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables |
| | Business Case Project Charter Detailed Business Reqmt's | High Level Solution Design Testing Strategy | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria | Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria |
| | Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

Project Management with SDLC

Initial Project Planning



Internal Dynamics - SDLC Interfaces



- ▶ **Project Management Lifecycle**

- **Initial Project Planning (Project Plan and Project Schedule)**
- Ongoing Project Execution (Issue Log, Risk Log, Change Request, Status)

- ▶ **Software Testing Lifecycle**

- Unit Test Planning, Execution & Authorization

Internal Dynamics - SDLC Interfaces



► Project Management Lifecycle

- Initial Project Planning (Project Plan and Project Schedule)
- **Ongoing Project Execution (Issue Log, Risk Log, Change Request, Status)**

► Software Testing Lifecycle

- Unit Test Planning, Execution & Authorization

Project Management with SDLC

Ongoing Project Execution



QAassist Integrated Methodology

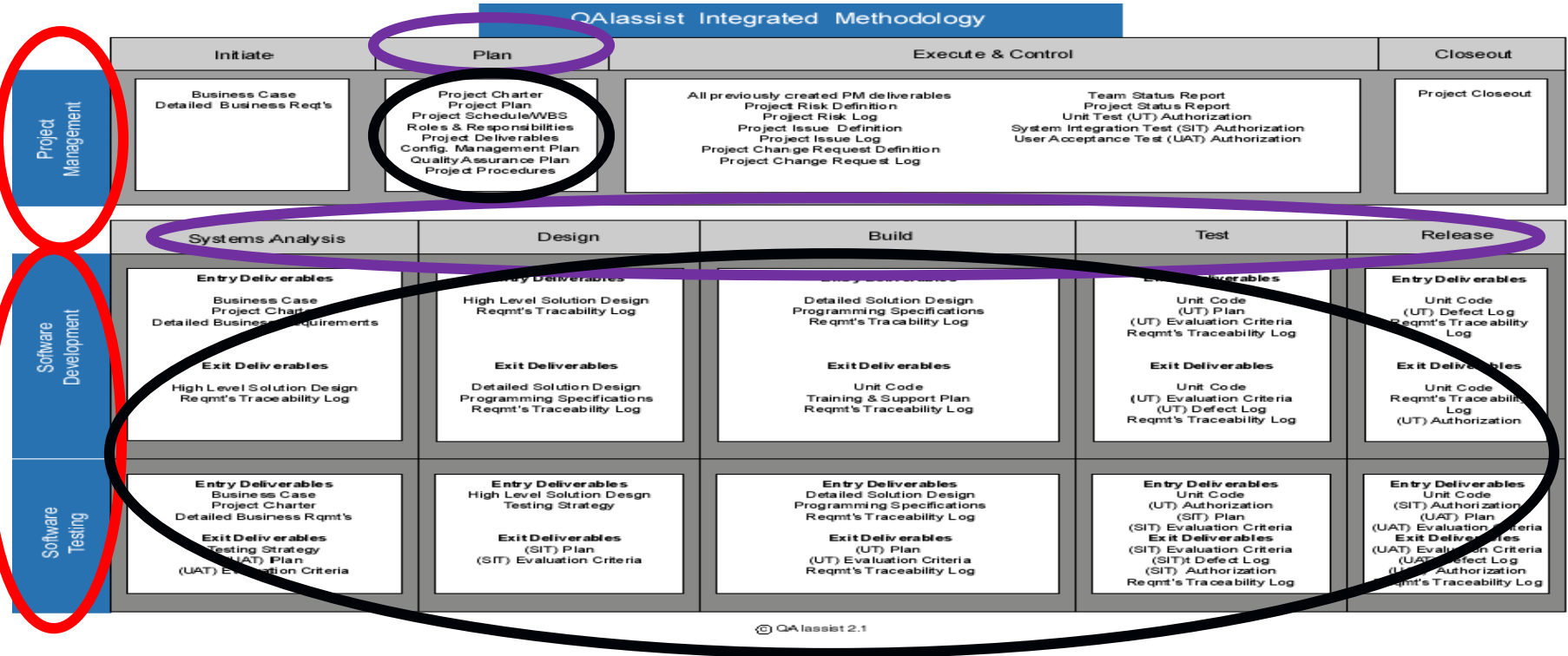
| | QAassist Integrated Methodology | | | | |
|--------------------|--|--|---|--|--|
| | Initiate | Plan | Execute & Control | | Closeout |
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| | Systems Analysis | Design | Build | Test | Release |
| | Software Development | Software Development | Software Development | Software Development | Software Development |
| | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables |
| | Business Case Project Charter Detailed Business Requirements | High Level Solution Design Reqmt's Traceability Log | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Unit Code (UT) Defect Log Reqmt's Traceability Log |
| | Exit Deliverables | Exit Deliverables | Exit Deliverables | Exit Deliverables | Exit Deliverables |
| | High Level Solution Design Reqmt's Traceability Log | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code Training & Support Plan Reqmt's Traceability Log | Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Unit Code Reqmt's Traceability Log (UT) Authorization |
| | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables |
| | Business Case Project Charter Detailed Business Reqmt's | High Level Solution Design Testing Strategy | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria | Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria |
| | Exit Deliverables | Exit Deliverables | Exit Deliverables | Exit Deliverables | Exit Deliverables |
| | Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | (SIT) Plan (SIT) Evaluation Criteria | (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |

Project Management with SDLC

Ongoing Project Execution

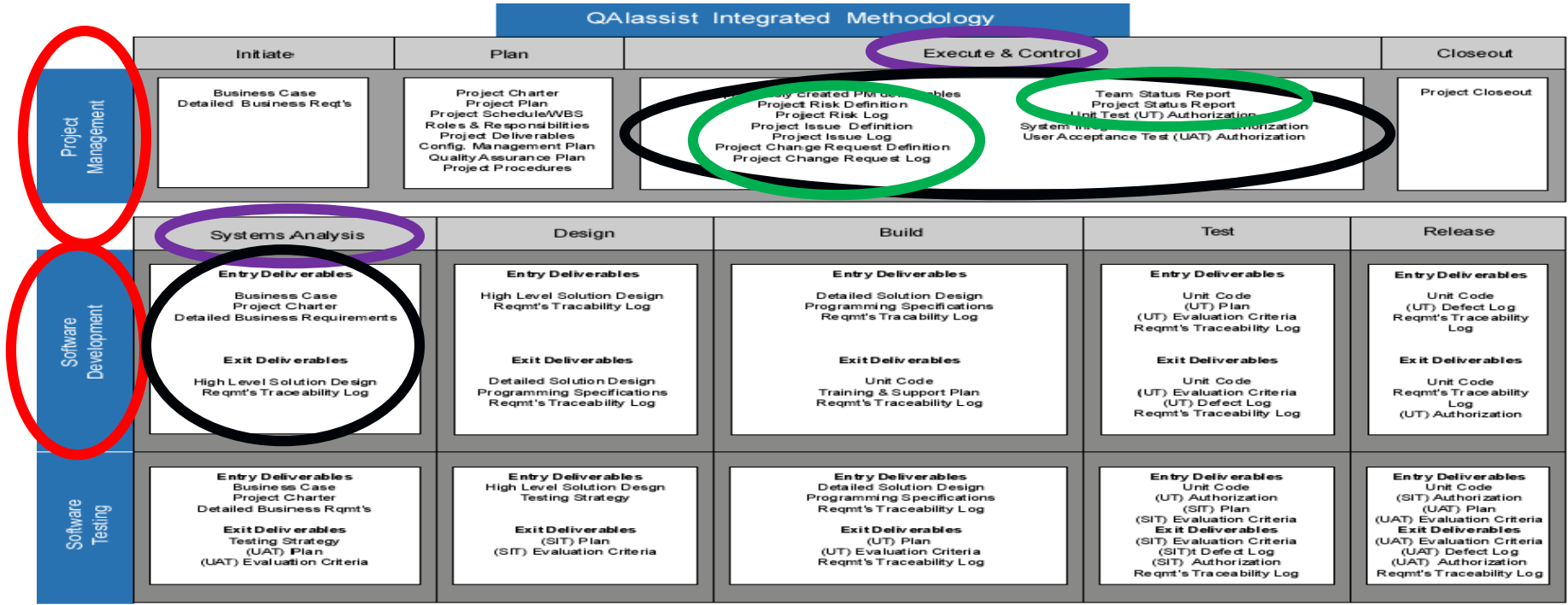


QAassist Integrated Methodology



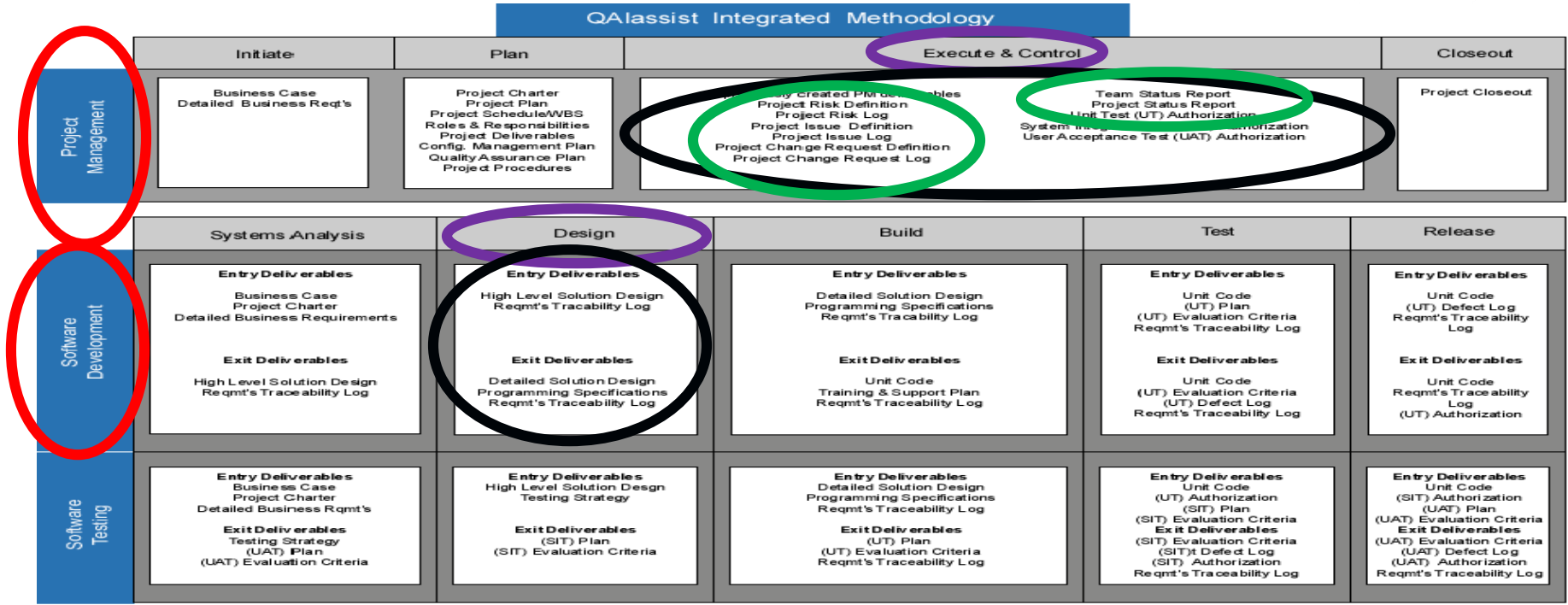
Project Management with SDLC

Ongoing Project Execution



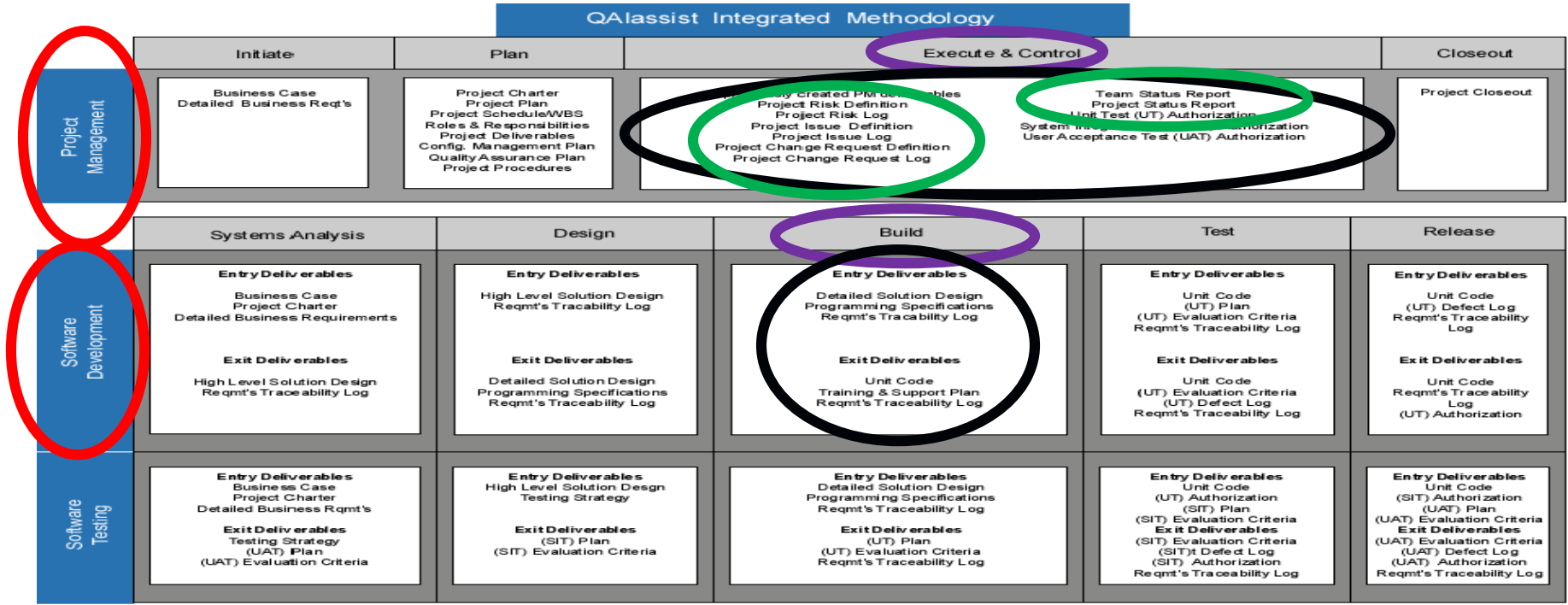
Project Management with SDLC

Ongoing Project Execution



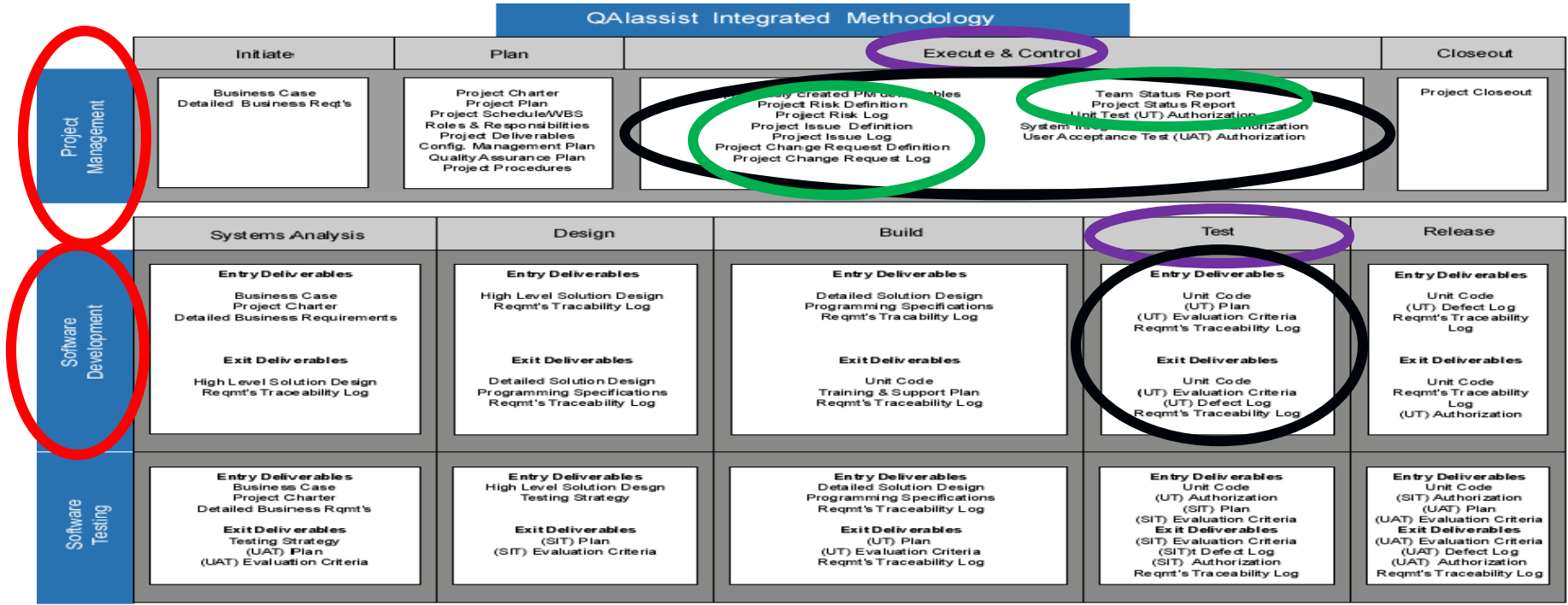
Project Management with SDLC

Ongoing Project Execution



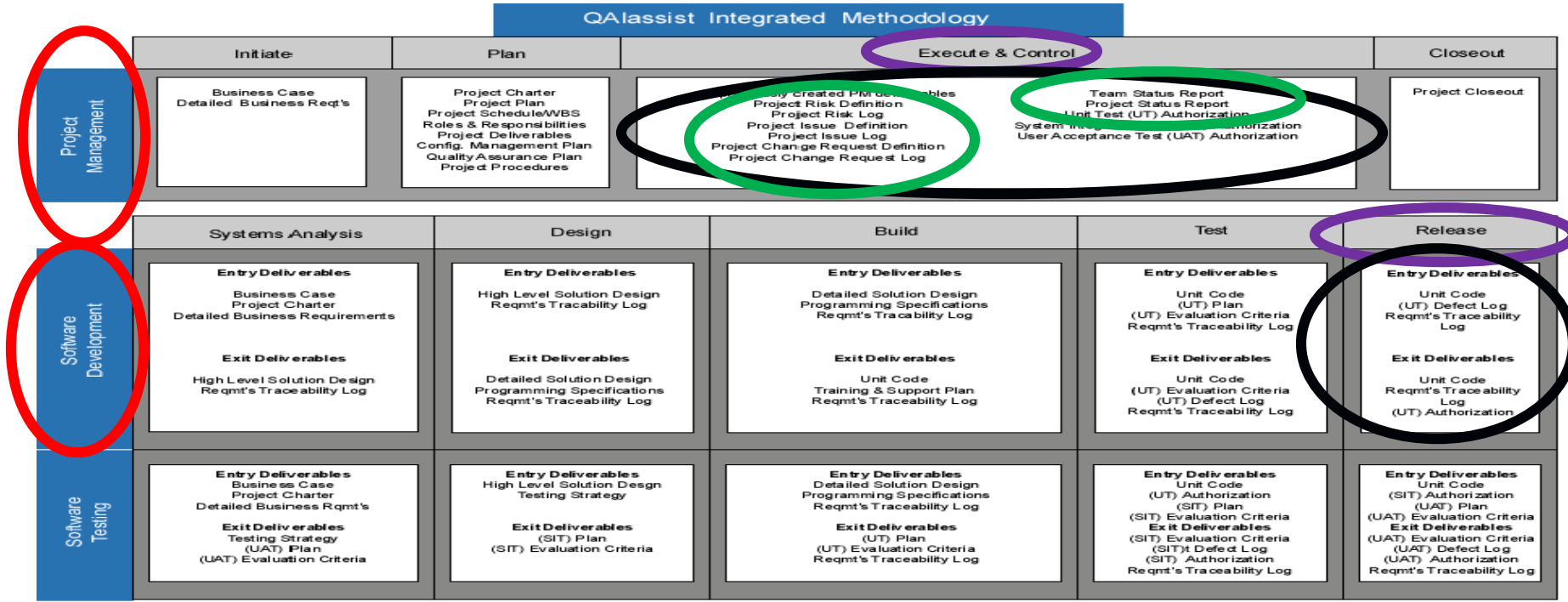
Project Management with SDLC

Ongoing Project Execution



Project Management with SDLC

Ongoing Project Execution



Internal Dynamics - SDLC Interfaces



► Project Management Lifecycle

- Initial Project Planning (Project Plan and Project Schedule)
- **Ongoing Project Execution (Issue Log, Risk Log, Change Request, Status)**

► Software Testing Lifecycle

- Unit Test Planning, Execution & Authorization

Internal Dynamics - SDLC Interfaces



- ▶ **Project Management Lifecycle**

- Initial Project Planning (Project Plan and Project Schedule)
- Ongoing Project Execution (Issue Log, Risk Log, Change Request, Status)

- ▶ **Software Testing Lifecycle**

- Unit Test Planning , Execution & Authorization



SDLC with Software Testing Lifecycle (STLC)

Unit Test Planning, Execution & Authorization

QAassist Integrated Methodology

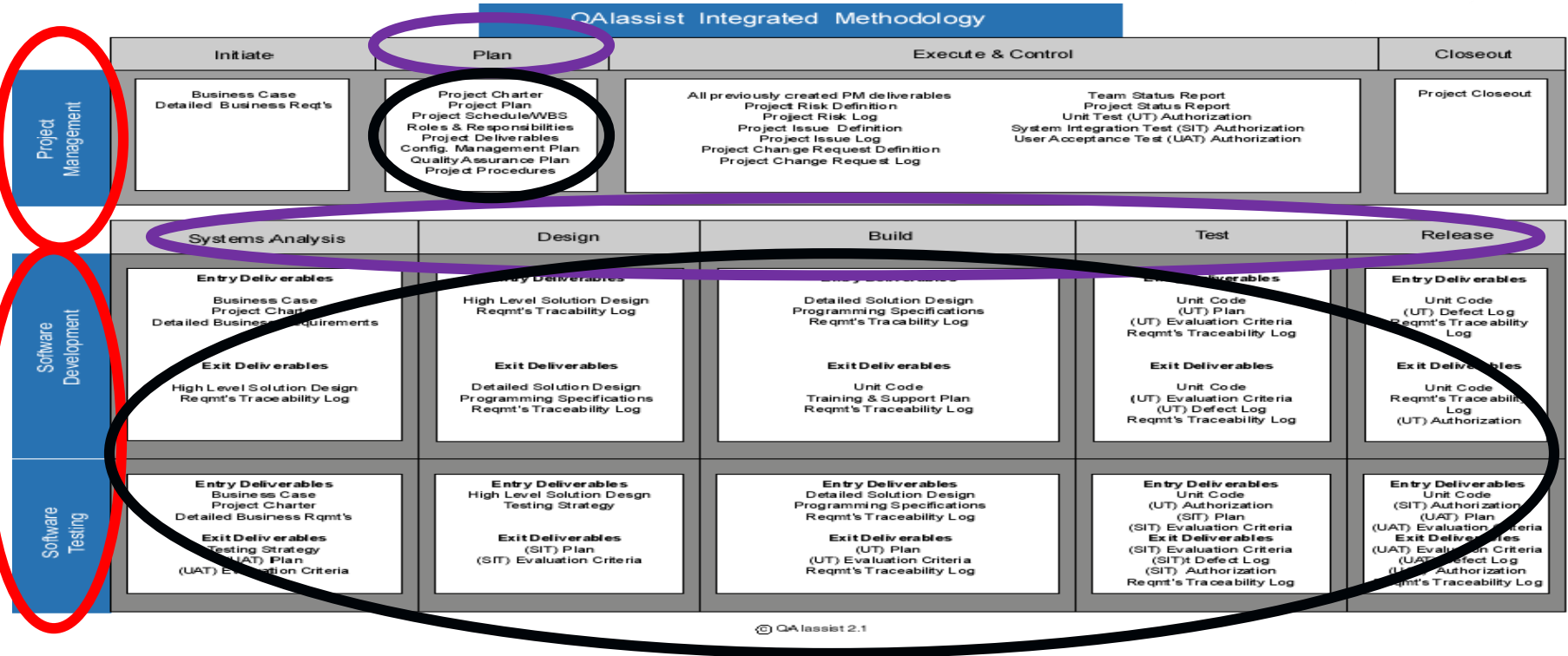
| | QAassist Integrated Methodology | | | | |
|----------------------|--|--|---|---|--|
| | Initiate | Plan | Execute & Control | | Closeout |
| Project Management | Business Case Detailed Business Req'ts | Project Charter Project Plan Project Schedule/VBS Roles & Responsibilities Project Deliverables Config. Management Plan Quality Assurance Plan Project Procedures | All previously created PM deliverables Project Risk Definition Project Risk Log Project Issue Definition Project Issue Log Project Change Request Definition Project Change Request Log | | Team Status Report Project Status Report Unit Test (UT) Authorization System Integration Test (SIT) Authorization User Acceptance Test (UAT) Authorization |
| | Systems Analysis | Design | Build | Test | Release |
| | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables |
| Software Development | Business Case Project Charter Detailed Business Requirements | High Level Solution Design Reqmt's Traceability Log | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Unit Code (UT) Defect Log Reqmt's Traceability Log |
| | Exit Deliverables High Level Solution Design Reqmt's Traceability Log | Exit Deliverables Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Exit Deliverables Unit Code Training & Support Plan Reqmt's Traceability Log | Exit Deliverables Unit Code (UT) Evaluation Criteria (UT) Defect Log Reqmt's Traceability Log | Exit Deliverables Unit Code Reqmt's Traceability Log (UT) Authorization |
| Software Testing | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables | Entry Deliverables |
| | Business Case Project Charter Detailed Business Reqmt's | High Level Solution Design Testing Strategy | Detailed Solution Design Programming Specifications Reqmt's Traceability Log | Unit Code (UT) Authorization (SIT) Plan (SIT) Evaluation Criteria | Unit Code (SIT) Authorization (UAT) Plan (UAT) Evaluation Criteria |
| | Exit Deliverables Testing Strategy (UAT) Plan (UAT) Evaluation Criteria | Exit Deliverables (SIT) Plan (SIT) Evaluation Criteria | Exit Deliverables (UT) Plan (UT) Evaluation Criteria Reqmt's Traceability Log | Exit Deliverables (SIT) Evaluation Criteria (SIT) Defect Log (SIT) Authorization Reqmt's Traceability Log | Exit Deliverables (UAT) Evaluation Criteria (UAT) Defect Log (UAT) Authorization Reqmt's Traceability Log |



SDLC with Software Testing Lifecycle (STLC)

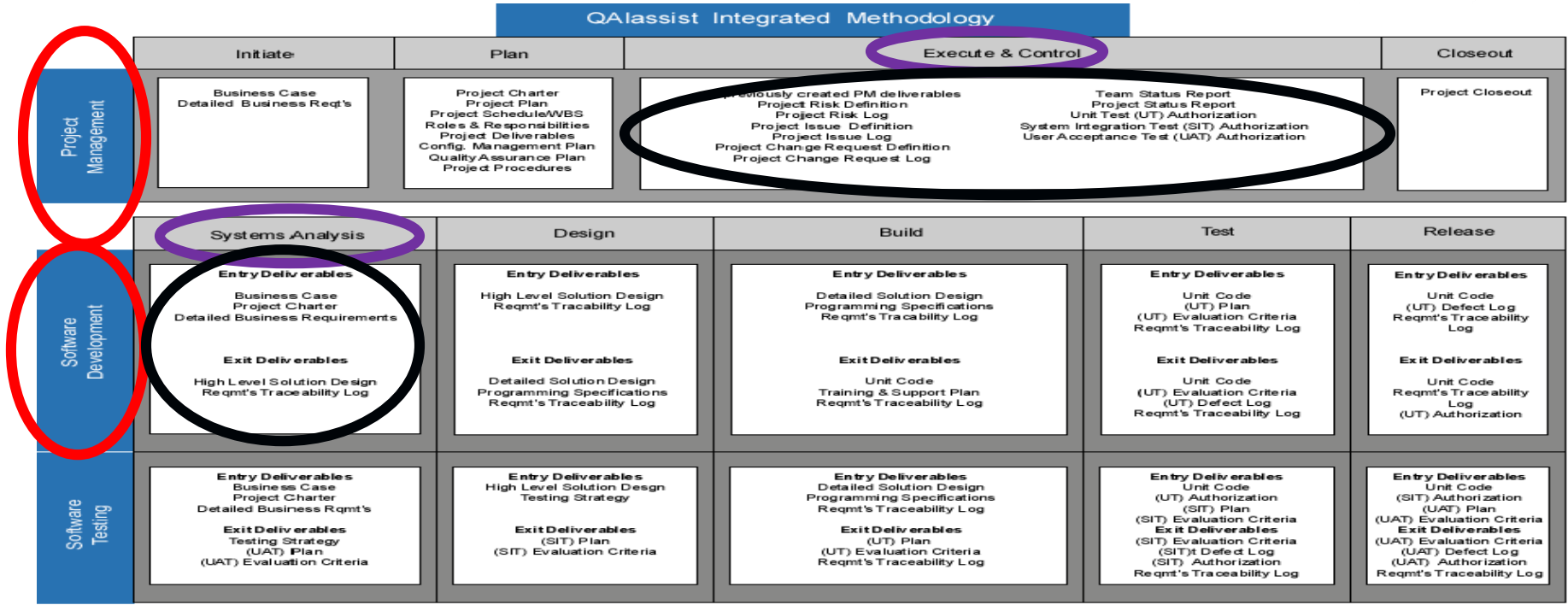
Unit Test Planning, Execution & Authorization

QAassist Integrated Methodology



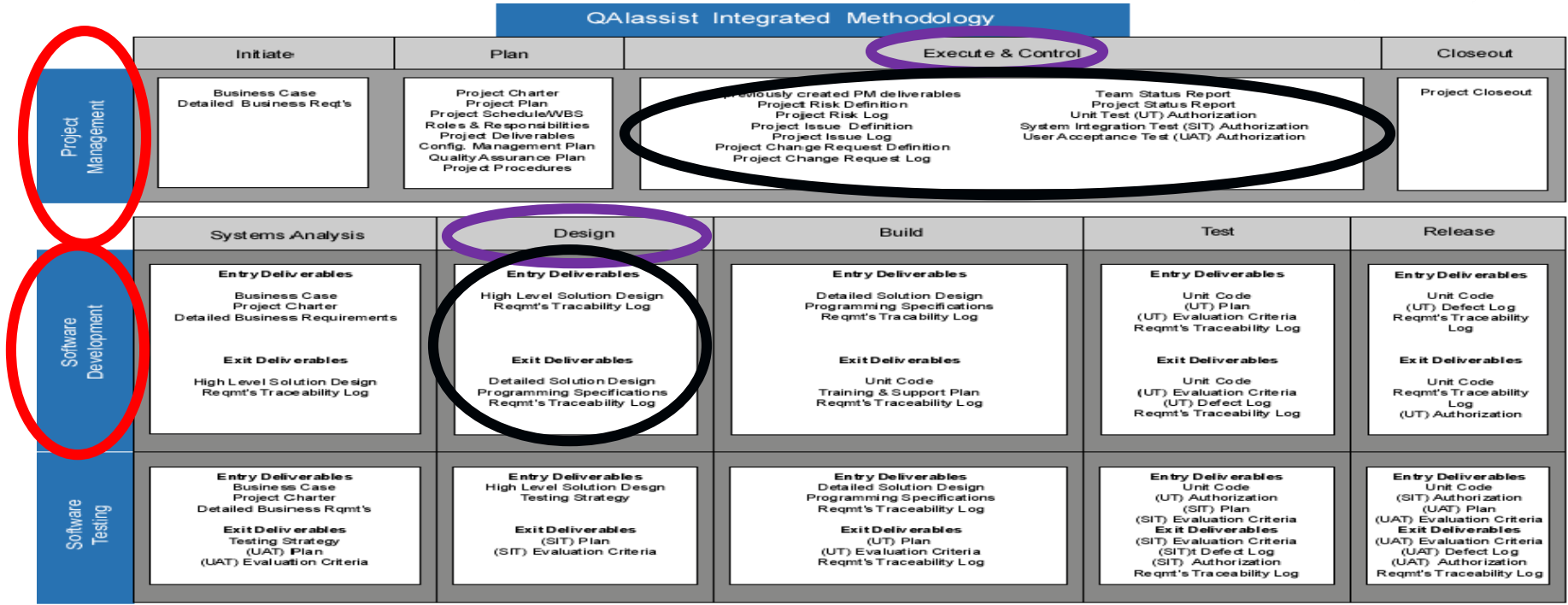
SDLC with Software Testing Lifecycle (STLC)

Unit Test Planning, Execution & Authorization



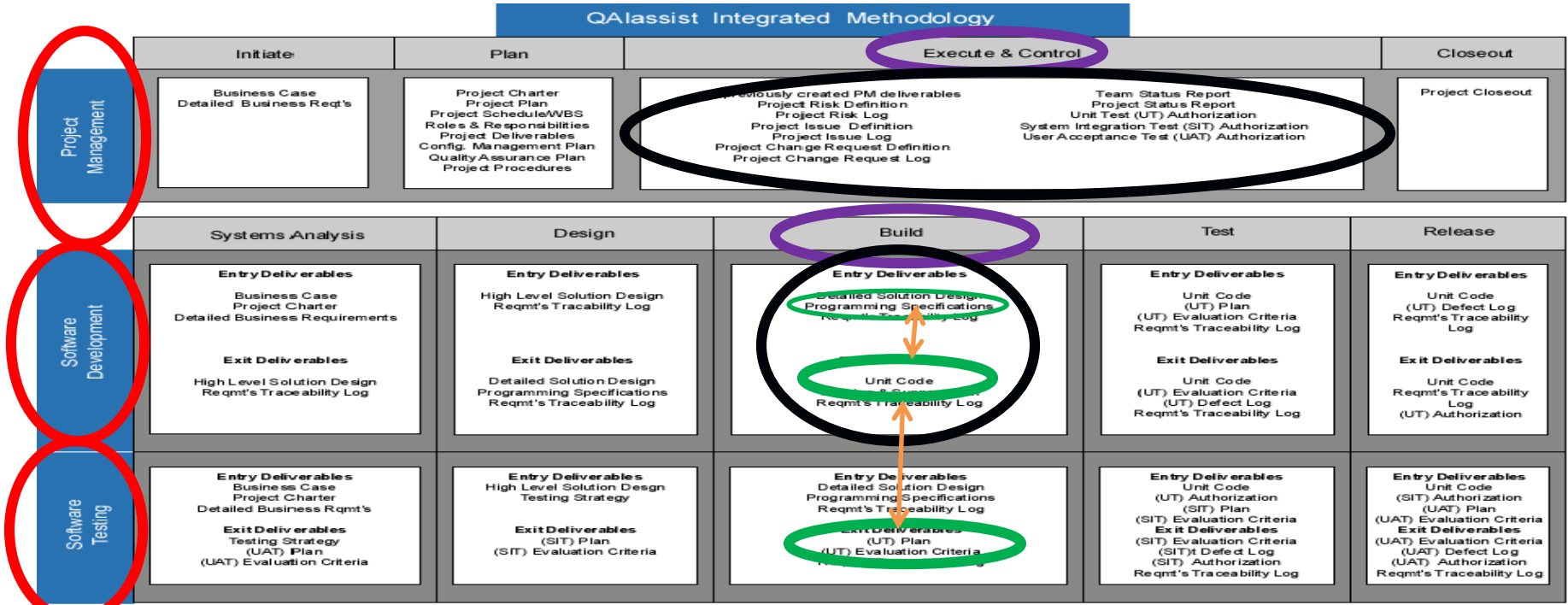
SDLC with Software Testing Lifecycle (STLC)

Unit Test Planning, Execution & Authorization



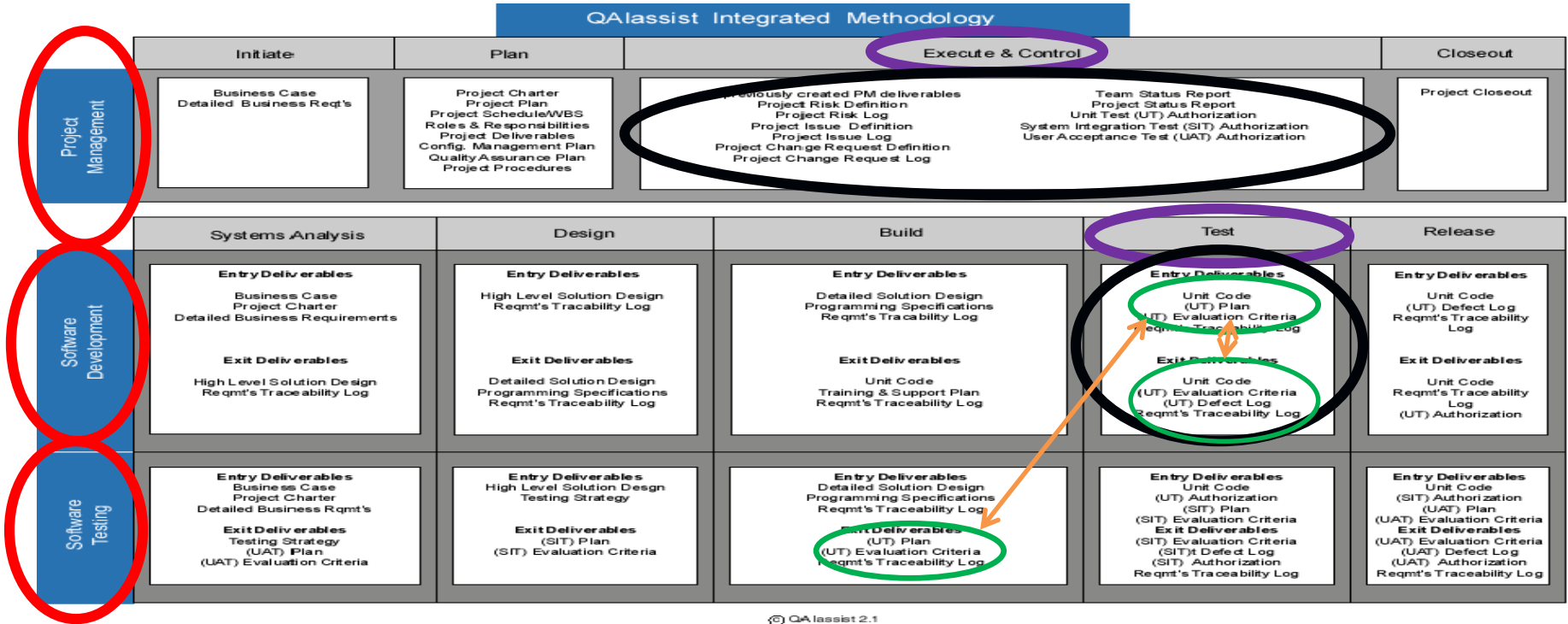
SDLC with Software Testing Lifecycle (STLC)

Unit Test Planning, Execution & Authorization



SDLC with Software Testing Lifecycle (STLC)

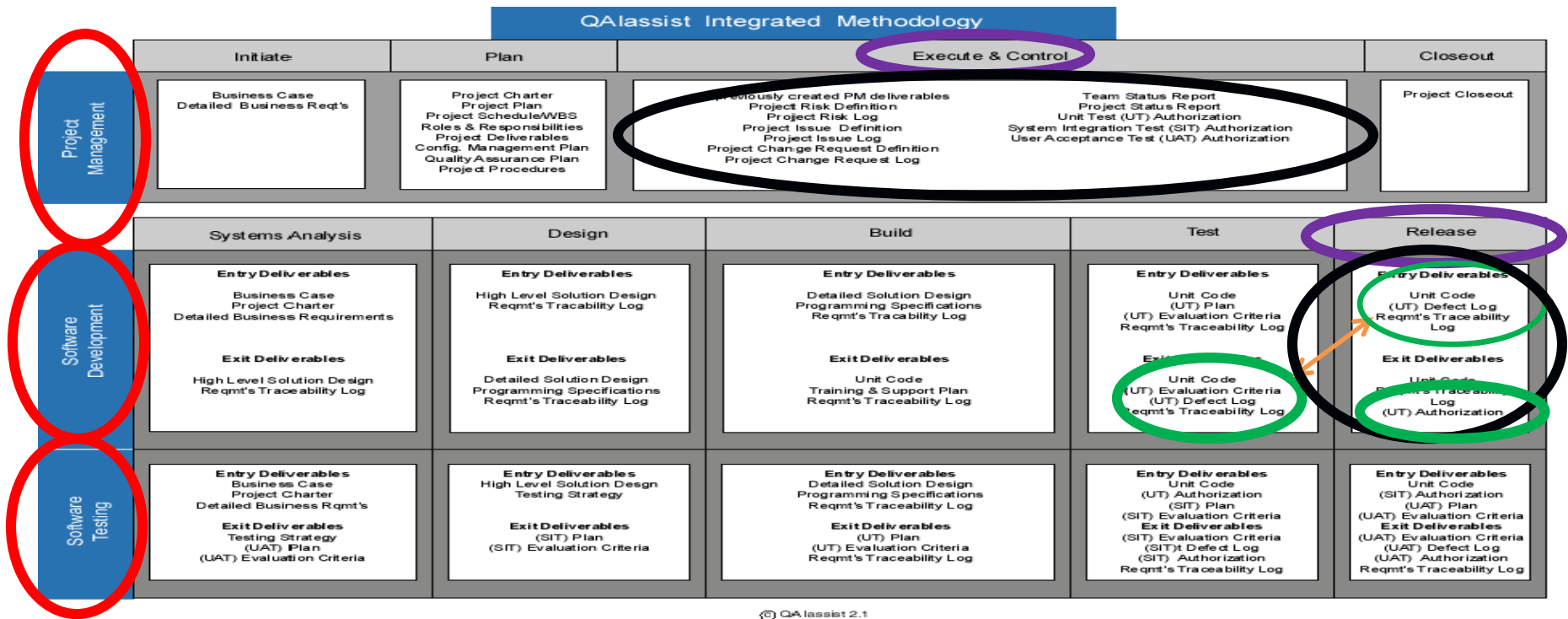
Unit Test Planning, Execution & Authorization





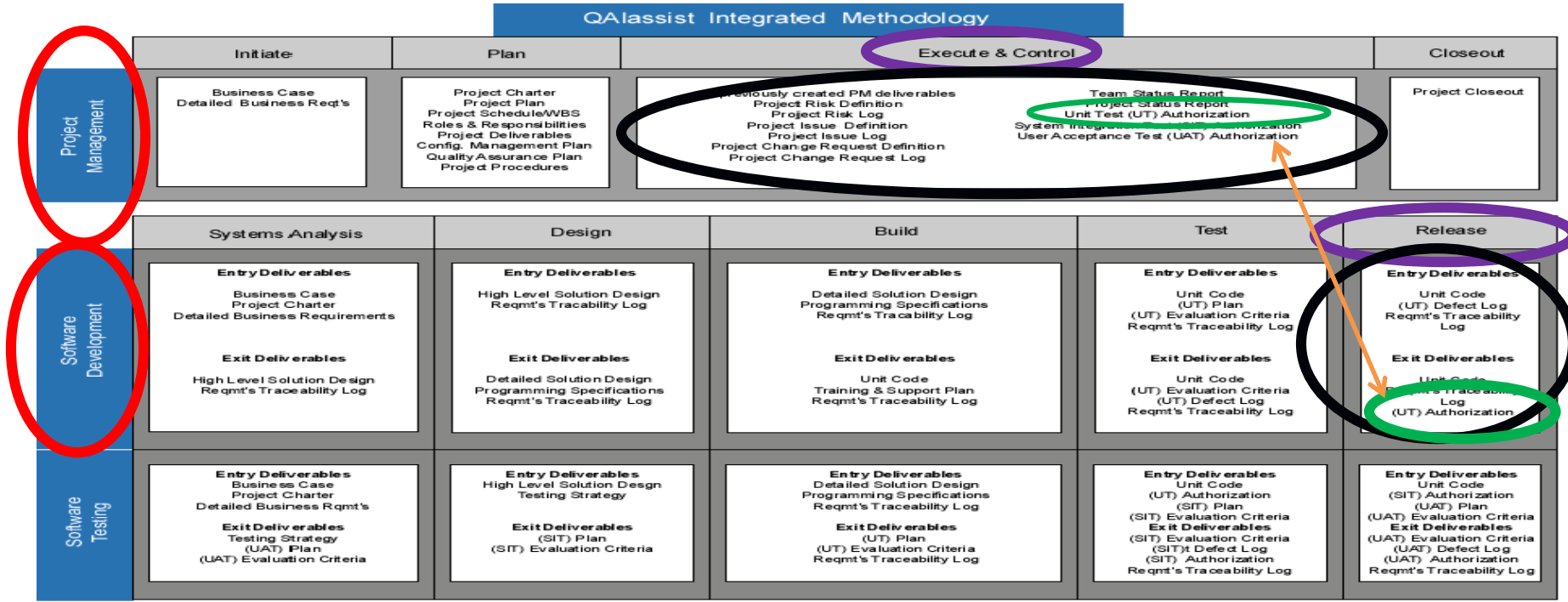
SDLC with Software Testing Lifecycle (STLC)

Unit Test Planning, Execution & Authorization



Project Management with SDLC

Unit Test Authorization



Leverage the Software Development Lifecycle

Concepts III - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)**
- Recap

Leverage the Software Development Lifecycle ConceptsIII - Agenda



- Establish webinar context - review previous “Concepts” sessions
 - Jan - A Context for IT Methodology
 - Mar- Apply PM Fundamentals to IT
- Software Development Lifecycle (SDLC) – Context and Premise
- SDLC - Requirements (Identification and Definition)
- SDLC Phases - Objectives, Iterations, Traceability
- SDLC – Internal Dynamics (PMLC and STLC interfaces)
- **Recap**

Body of Knowledge



- ▶ Methodology/Lifecycle/Deliverables – www.qaiassist.com
 - Project Management, Software Development, Software Testing
- ▶ Body of Knowledge (Library) – www.qaiassist.com
 - **Methodology Origin** – origins of methodology – identifies the differences between the path we take (noun) and how we proceed (verb) down that path
 - **Context & Overview** – provides a general overview of IT Methodology and how it can be applied within an organization
 - **Deliverable Descriptions** - provides general descriptions for all the deliverables of the QAIassist Integrated Methodology

Until Next Session...



- **Keep Posing your Questions** – call or send your questions in (Denise and/or Cameron)
- **Explore the QAlassist Body of Knowledge** – refer to the QAlassist Body of Knowledge (sample deliverables, whitepapers, blog articles) – ([www-qaiassist.com](http://www.qaiassist.com))
- **QAlassist Practitioner** - find out more about becoming QAlassist certified “Foundation”, “Practitioner”
- **Tell your friends** - tell your friends, coworkers and colleagues about the Project Insight/QAlassist webinars – invite them to join us at the next session
- **Continuing Follow Up** – send out a survey at the completion of every webinar, follow up with attendees to obtain their suggestions and recommendation



2016 Schedule



- **The 3rd Wednesday of Every Month**

- 8:00 am Pacific - 11:00 am Eastern Time

- **IT Methodology Concepts (60 mins)**

- This series of six webinars will be presented on a bi-monthly basis. Each webinar will provide you with a context, overview, terminology and general understanding of information technology (IT) methodology.

- **IT Methodology – Tutorials (60 mins)**

- This series of five webinars will be presented on a bi-monthly basis. Each session is designed to offer a more “hands on” perspective – bridging the concepts into a practical utilization.



2016 “Concepts” Schedule



- ▶ **The 3rd Wednesday of Month (Jan, Mar, May, Jul, Sept, Nov)**
 - 8:00 am Pacific - 11:00 am Eastern Time

- ▶ **IT Methodology Concepts (60 mins)**
 - Methodology for the IT Landscape - (Jan)
 - Apply PM Fundamentals to IT - (Mar)
 - Leverage the Software Development Lifecycle - (May)
 - **Explore the Software Testing Lifecycle - (Jul)**
 - Incorporate Organizational Process Governance - (Sept)
 - Implementing IT Methodology - (Nov)

2016 “Tutorials” Schedule



- **The 3rd Wednesday of Month (Feb, Apr, Jun, Aug, Oct)**

- 8:00 am Pacific - 11:00 am Eastern Time

- **IT Methodology Tutorials (60 mins)**

- Applying IT Methodology to Project Initiation - (Feb)
- Project Planning & Design with IT in Mind - (Apr)
- **Executing Projects with IT Methodologies - (Jun)**
- Project Control & Verification - (Aug)
- Project Close & Delivery - (Oct)



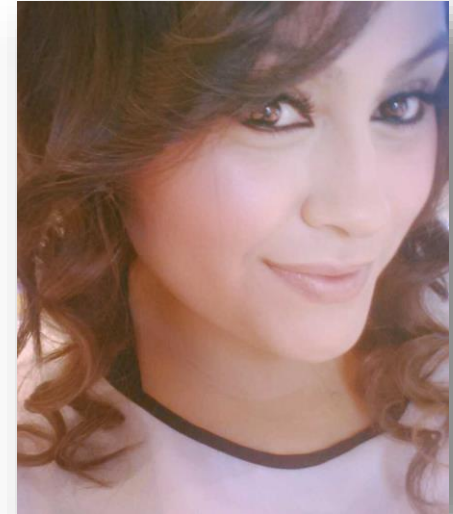
Denise Rodriguez

Project Insight

Marketing

Denise.Rodriguez@projectinsight.com

www.projectinsight.net



Join us!



Twitter



LinkedIn



Facebook



Instagram



YouTube



Learn more...



- Schedule a customized demo today
 - +1 (949) 476-6499 x3
 - info@projectinsight.net
 - Request info: www.projectinsight.net
- Contact QAlassist
 - +1 (613) 523-0052
 - solutions@qaiassist.com
 - Request info: www.qaiassist.com



Project Insight Community



- ▶ Sign up for more!
- ▶ IT and Agile Methodology
- ▶ PM Training
- ▶ Leadership
- ▶ Product Training
- ▶ Check out videos

