Quality Assurance
Defect Management Process

Initial roll out: Finance, Beazley Digital Platform

v0.4[DRAFT]
July 2022

beazley

beautifully
designed
insurance

Context

- This document outlines Beazley's Defect Management process and best practice to be adopted across all IT enabled change delivery. The approach outlined is designed to provide visibility of all defects across product(s), platform(s) / value stream(s) and enterprise level ensuring all defects receive the appropriate attention enabling E2E traceability and compliance to audit and regulatory requirements.

- A key factor in the successful completion of QA activities is a well-managed Defect Management process. This involves collecting and processing information on defects that occur during static and dynamic testing and analysing that information so that trends and significant events are recognised. There are six core objectives to Defect Management:

    1. Ensure that all defects are logged and tracked with minimum impact to delivery timescales;
    2. Analyse defects to reduce their impact to the project, task or service, and allow reporting on the current test status;
    3. Maintain centralised control and prioritise the fix and retest of defects;
    4. Facilitate the workflow of a defect from creation through to resolution;
    5. Facilitate root cause analysis of defects to inform continuous process improvement; and
    6. Ensure engagement of all relevant stakeholders.

- As part of the Jira standards work currently being undertaken, we will be standardising our Test and Defect Management processes to ensure consistency across Beazley platforms / value streams and improve delivery quality using an effective and optimised process. We will start with implementing Defect Management followed by Test Management and process improvements.

Key Changes

- To consolidate and simplify the various defect workflows we have, we will be moving to a single dedicated defect workflow with the following key changes:
  - Transition from ***Bug*** to ***Defect*** to keep naming convention consistent with test management toolset and reporting
  - Introducing New Statuses (pg5) to simply and standardise under one workflow
  - Introducing ***mandatory**** fields (existing and new) to allow for effective defect management reporting and drill down capability; this includes:
    - Platform
    - Product
    - Component
    - Build Version
    - Environment
    - Expected & Actual Results
    - Defect Severity & Defect Priority
    - Root Cause / Root Cause Description
    - Epic Link
  - Implement Jira workflow automation logic. Few sample automation rules as follows:
    - Close parent when all sub-tasks have been completed
    - Auto close parent issue when all sub-tasks are done

- *Note: the proposed mandatory fields above will be made mandatory to the ***Defect*** issue type only.

- **Next Steps:**
  - Andreas will migrate the existing **78 *open*** Beazley Digital Bugs to the new Defect Standard as per mapping sheet pgs7-11 commencing Fri 8th July. The 1086 Closed Bugs will also be migrated at a later date to enable historical reporting.
  - Andreas will migrate the existing **18 *open*** Finance Bugs to the new Defect Standard as per mapping sheet pgs7-11 commencing Fri 8th July. The 235 Closed Bugs will also be migrated at a later date to enable historical reporting.
  - Note: any existing custom dashboards or reports using the above will need to be remapped; there are a new set of standard dashboards which we will publish for your use.

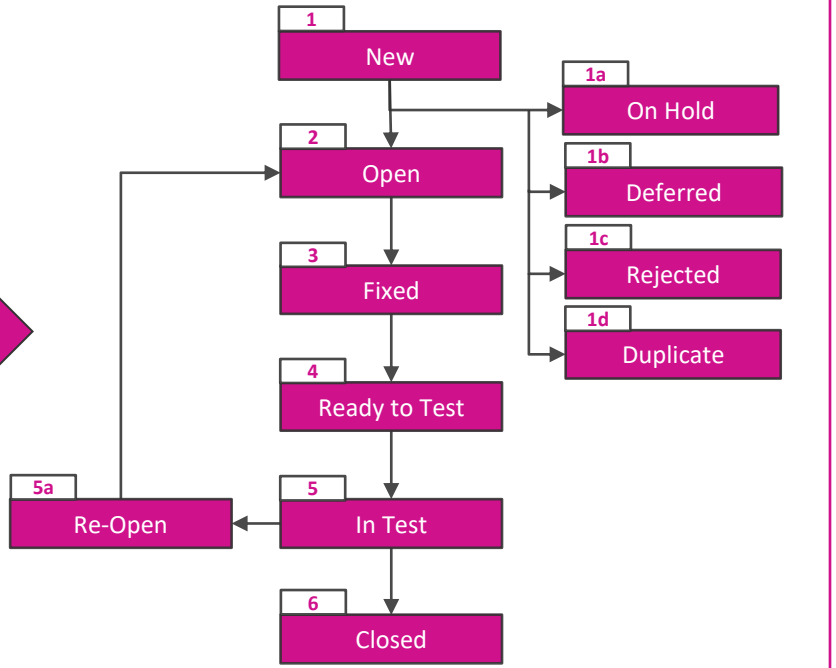
Jira Defect Workflow(s) – Current vs New

Digital BUG WORKFLOW

Finance / FMOD

Transition

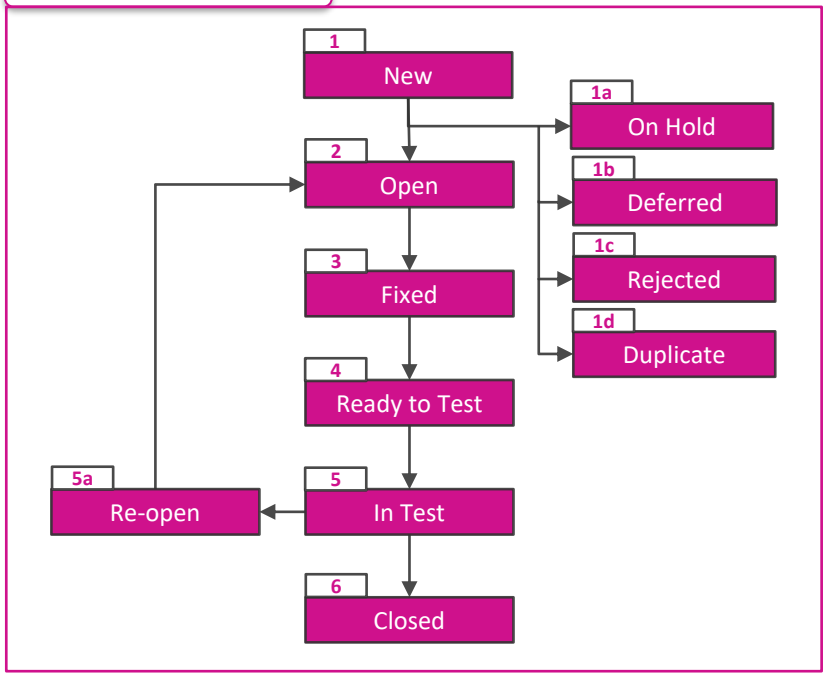
NEW DEFECT WORKFLOW



beazley

Defect Workflow Summary

NEW DEFECT WORKFLOW

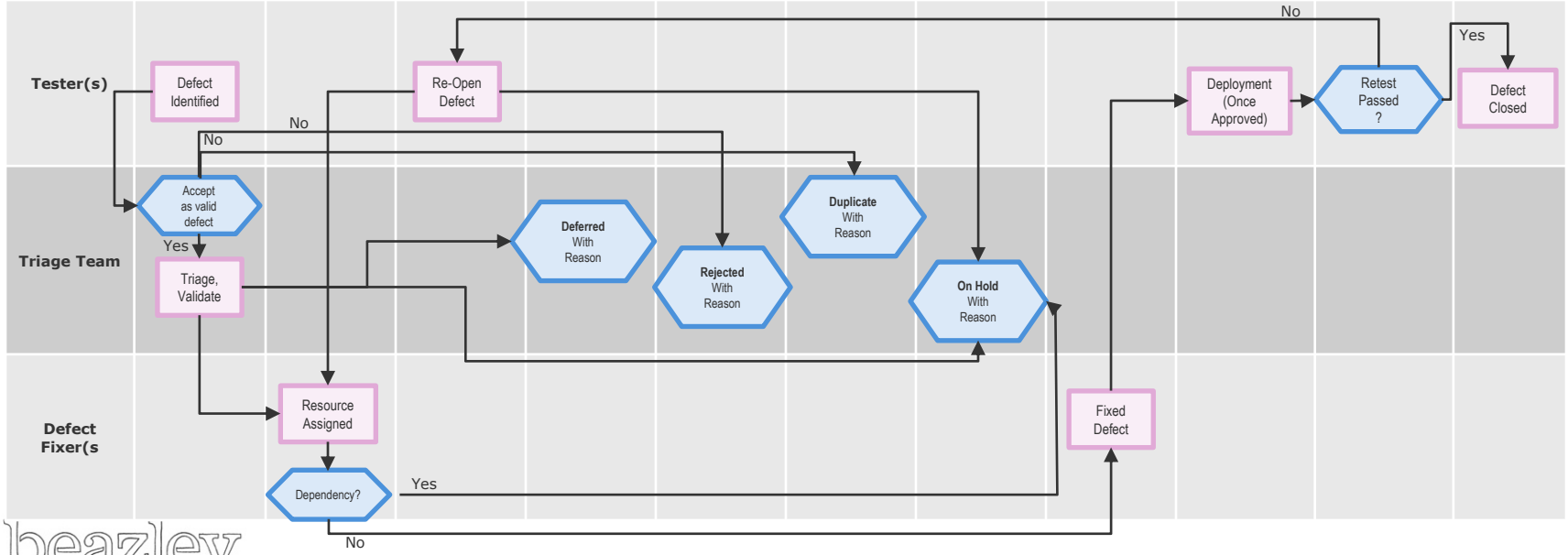


	Status	Process	Actioned By
1	New	Defect is identified. Triage is performed against the defect	Originator / Triage
1a	On Hold	If there is a dependency to work on the defect then the defect can be moved to On Hold	Triage
1b	Deferred	Triage team makes the decision whether the defect can be deferred	Triage
1c	Rejected	Triage team makes the decision whether the defect can be Rejected	Triage
1d	Duplicate	If the defect is already exists then the new defect is moved to Duplicate status	Triage
2	Open	Defect is assigned and status is changed to Open	Resolver
3	Fixed	Defect is repaired and unit tested	Resolver
4	Ready to Test	Once approved the build is promoted to QA environment	Resolver
5	In Test	Defect is tested and compared with expected results	Testers
5a	Re-Open	When the Retest fails the defect is Re-opened	Testers
7	Closed	Defect is Resolved and validated by Originator	Originator

Triage team consists of BA, Developers, Config and testers

Defect Workflow Process

Status	New	Open	Re-open	Deferred	Rejected	Duplicate	On Hold	Fixed	Ready To Test	In Test	Closed
Process	Defect is identified. Triage is performed against the defect	Defect is assigned and status is changed to Open	When the Retest fails the defect is Re-opened	Triage team makes the decision whether the defect can be deferred	Triage team makes the decision whether the defect can be Rejected	If the defect is already exists then the new defect is moved to Duplicate status	If there is a dependency to work on the defect then the defect can be moved to On Hold	Defect is repaired and unit tested	Once approved the build is promoted to QA environment	Defect is tested and compared with expected results	Defect is Resolved and validated by Originator



beazley

Bug > Defect Mapping

BOLD	Mandatory field(s)
	New Field
	Exists in BUG and not in DEFECT

DEFECT	BUG	Type	BUG Values	DEFECT Values	Mapping / Suggestion
Platform	Platform	Drop Down			As per configuration
Product	Product	Drop Down			As per configuration
Product (BZD)	Product (BZD)	Drop Down			As per configuration
Project	**Project**	Drop Down			As per configuration
Component	Component	Drop Down			As per configuration
Sub-Components	Sub-Components	Drop Down			As per configuration
Build Version		Free Text			Build version needs to be entered manually now. Once Octopus Deploy is integrated with Jira this will be available in drop down
Summary	**Summary**	Free Text			
Environment	Environment	Drop Down			As per current configuration
Description	**Description**	Free Text			
Expected Results	Expected Results	Free Text			
Actual Results	Actual Results	Free Text			
Test Data		Free Text			Test Data details needs to be entered manually in here as free text
Defect Priority	Priority	Drop Down	Highest High Medium Low Lowest Normal Minor Trivial Moderate Critical Major Blocker	P1 – Blocker P2 – Major P3 – Medium P4 – Minor P5 - Trivial	Critical, Highest - P1 Blocker High - P2 Major Moderate - P3 Medium Normal, Low - P4 Minor Lowest - P5 Trivial
Defect Severity	Severity	Drop Down	Severity 1 – No Workaround Severity 2 – Regulatory or Financial Severity 3 – Workaround available Severity 4 – Small Change Request Severity 5 – Backlog Item	S1 – Critical S2 – Major S3 – Medium S4 – Low S5 – Minor	Severity 1 to S1 ..... Severity 5 to S5

Bug > Defect Mapping

BOLD	Mandatory field(s)
	New Field
	Exists in BUG and not in DEFECT

DEFECT	BUG	Type	BUG Values	DEFECT Values	Mapping / Suggestion
Reporter	Reporter	Auto			
	Reporter (Business)	Drop Down			New non mandatory field
Assignee	Assignee				
Test ID		Free Text			To link defect to Test - once Xray is implemented the test id will be populated automatically from test execution.
Linked Issues	Linked Issues	Drop Down			
Label	Label				
Sprint	Sprint	Drop Down			
Attachment	Attachment				
Fix Version	Fix Version	Drop Down			
Root Cause		Drop Down		Software Design Requirements Infra/Environment Test Other	When the defect is moved to Fixed from Open, this field will become mandatory to the developer
Root Cause Description		Drop Down		**Software** Code – lack of unit / component test Code Compatibility Coding Error Code - Lack of Code Code Pattern- error in pattern (base error) Failure to Adhere to effective code principles / standards Failure to initialise or check data values Inadequate error detection and recovery Incorrect algorithm or computation Incorrect branch or loop Interface problem between module or program	Once the Root Cause is selected this field will be enabled

Bug > Defect Mapping

BOLD	Mandatory field(s)
	New Field
	Exists in BUG and not in DEFECT

DEFECT	BUG	Type	BUG Values	DEFECT Values	Mapping Suggestion
Root Cause Description		Drop Down		unnecessary embellishment Code – Data Configuration issue Code - Performance Issue Database issue (eg., Indexing) Software Configuration issue / Configuration – Application **Design** Design compatibility Design/standards not met Failure to fulfil design Failure to meet stated requirement Inadequate error detection and recovery Incorrect algorithm or logic Incorrect data structure or representation Interface design problem between components or subsystem Over engineering or unnecessary System resource utilization or inefficiency problem Requirements Ambiguous, non unique, non testable Incorrect assumption Missing Requirement Superfluous requirement Incorrect Requirement Unclear Requirement **Requirements** Ambiguous, non unique, non testable Incorrect assumption Missing Requirement Superfluous requirement Incorrect Requirement Unclear Requirement	Once the Root Cause is selected this field will be enabled

Bug > Defect Mapping

BOLD	Mandatory field(s)
	New Field
	Exists in BUG and not in DEFECT

DEFECT	BUG	Type	BUG Values	DEFECT Values	Mapping Suggestion
Root Cause Description		Drop Down		**Infra / Environment** Data quality issue Build failure Hardware Configuration issue Recovery / exception issue Security issue Start up / restart services issue Workload / stress issue Environment Issue Rulebook Issue Access rights issue Performance Issue **Test** Incorrect test condition / test case Incorrect execution Incorrect data Invalid defect **Other** Business rule Duplicate defect Legacy / existing system or feature Not Reproducible Out of Scope Typo Reference Data Known Error / Bug - Application	Once the Root Cause is selected this field will be enabled

Bug > Defect Mapping

BOLD	Mandatory field(s)
	New Field
	Exists in BUG and not in DEFECT

DEFECT	BUG	Type	BUG Values	DEFECT Values	Mapping / Suggestion
Reason		Free text			When the defect is moved from Open to Deferred, Duplicate, On Hold or Rejected this field will become mandatory
Defect Type		Drop Down		Functional Design Requirements Infra/Environment Test Other Non-Functional Regression Automation	The Defect Type field is new and the values should be selected based on the test that the tester is performing
Team	Team	Drop Down			As per configuration
EPIC LINK	EPIC Link	Drop Down			As per configuration
	WorkType		Change BAU BAU - small change		This will remain as-is for User Stories. Not required for Defect.
	Story Points				This will remain as-is for User Stories. Not required for Defect.
	Start Date				This will remain as-is for User Stories. Not required for Defect.
	Due Date				This will remain as-is for User Stories. Not required for Defect.

Defect Severity & Priority

Defect Severity / Priority descriptions		
Defect Level	**Severity**	**Descriptions**
Level 1 (S1)	**Critical**	The defect causes failure of a critical business process, critical functionality or critical data within a software or hardware system or sub-system (severe enough to prevent a system/sub-system or release going live). The defect does not have a workaround and must be fixed immediately. A defect that creates a security risk – even if the end users do not see these.
Level 2 (S2)	**Major**	Defect that presents a systemic problem that prevents a core business process from completing with sufficient tolerances and/or affects core functionality or data for which there is no acceptable workaround.
Level 3 (S3)	**Medium**	Defect that does not have a significant adverse impact on the user experience and/or presents a problem impacting a core business process, functionality or data for which there is an acceptable workaround.
Level 4 (S4)	**Low**	Purely cosmetic (e.g. screen layout labelling etc. and/or has minimal impact on business processing either because little system functionality is affected else there is a good workaround for it.
Level 5 (S5)	**Minor**	Minor problem that will not be evident to the end user.

Defect Priority / description		
Defect Level	**Priorities**	**Descriptions**
Level 1 (P1)	**Blocker**	Critical defect with testing blocked or significant impact to test schedule / release. Urgent fix required within 1 day.
Level 2 (P2)	**Major**	Very high priority defect effecting test execution; Urgent fix/ETA required.
Level 3 (P3)	**Medium**	Priority defect – must be fixed ASAP as part of acceptance tests / exit criteria
Level 4 (P4)	**Minor**	Low priority defect – generally cosmetic (Minor) or low business impact. Effort to made to fix and include in next sprint / release though not impacting acceptance tests / exit criteria
Level 5 (P5)	**Trivial**	Fix on best endeavours basis.

The priority of a defect indicates the urgency of a defect to the testers and the impact on the rest of testing. The tester raising the defect will assign a priority which may change after further investigation.

Proposed Component(s)

The following list of components will need to be managed centrally to ensure consistency in labelling and reflect any changes:

BZD (to be reviewed)	Finance		
Software Reporting Referral Task Process Other Integration DRC Testing Metrics Data Automation Access And Identity	Agresso FTS LES db FDM ADM TDM WPS Upgrade FIS get Paid Imburse Prevaro Autorek Concur OneSum X QMA	Finance IFRS WPS (Acturial) BdxHighGear E2E Rec Policy Admin system Legacy Policy Admin system Claim Center Alteryx/Xceptor PowerBI (Power Platform) AlphaTax Premium Pro AuditBoard	FCF - Financial Ctrl Framework ERP EPM Treasury Management Billing system Disclosure Management Payroll systems

Proposed Component(s)

The following list of components will need to be managed centrally to ensure consistency in labelling and reflect any changes:

MyBeazleyUS					
PRODUCT	**COMPONENT**	**PRODUCT**	**COMPONENT**	**PRODUCT**	**COMPONENT**
PRODUCT	BBR v1 BBR v1 Documents BBR v1 Schemes BBR v1 DS Clearance	PATTERN	Client Pattern Beazley US Pattern Beazley US Pattern Data Beazley US Pattern Data Services Beazley US Pattern Local Sequel Patterns Rulebook Quoting Website Rulebook Service Website Rulebook Data Services Rulebook Servers Rulebook Author Rulebook Authoring Server Rulebook Maintenance Website Package Installer Rulebook Core Platform Rulebook Custom Functions Rulebook Patterns	INTEGRATONS	Insight - MSA CRM BZWIN**TBC FTS LES Agresso BI Claims Center**TBC AWS Release Pipeline

Proposed Component(s)

The following list of components will need to be managed centrally to ensure consistency in labelling and reflect any changes:

MyBeazley US/UK/ROW

PLATFORM	PRODUCT	COMPONENT
MyBeazley International	myBeazley Canada	ML Canada v1 ML Canada v1 Documents ML Canada v1 Schemes Clearance Kynd Lineage Navision
	myBeazley Singapore	SL Singapore v1 SL Singapore v1 Documents SL Singapore v1 Schemes
	myBeazley CML Spain	SL Spain v1 SL Spain v1 Documents SL Spain v1 Schemes
	myBeazley PI Germany	PI Germany v1 PI Germany v1 Documents PI Germany v1 Schemes
	myBeazley CML France	ML FR v1 ML FR v1 Documents ML FR v1 Schemes
	myBeazley CML UK	ML UK v1 ML UK v1 Service ML UK v1 Documents ML UK v1 Schemes

PLATFORM	PRODUCT	COMPONENT
MyBeazley International	Applied for all V2 products	SL Covers SL Covers Canada SL Covers French SL Covers German SL Covers Singapore SL Covers Spain SL Pattern SL Pattern Canada SL Pattern French SL Pattern German SL Pattern Singapore SL Pattern Spanish
	myBeazley PI UK	PI Product v1 PI UK v1 PI Bluefin v1 Documents PI UK v1 Documents PI UK v2 Documents PI UK v3 Documents PI UK v4 Documents PI UK v5 Documents PI UK v1 Schemes PI UK v1 Service
	myBeazley PI France	PI France v1 PI France v1 Documents PI France v1 Schemes

Proposed Component(s)

The following list of components will need to be managed centrally to ensure consistency in labelling and reflect any changes:

MyBeazley US/UK/ROW					
PLATFORM	**PRODUCT**	**COMPONENT**	**PLATFORM**	**PRODUCT**	**COMPONENT**
MyBeazley International	myBeazley MedMal	PeMedMal v1 PeMedMal v1 Documents PeMedMal v1 Schemes	MyBeazley International	Applied for all V2 \products	Client Pattern Rulebook Quoting Website Rulebook Service Website Rulebook Data Services Rulebook Servers Rulebook Author Rulebook Authoring Server Rulebook Maintenance Website Package Installer Rulebook Core Platform Rulebook Custom Functions Rulebook Patterns Insight Unirisx Bottomline Bottomline V-series Post Code FTS LES Agresso BI Claims Center Ceded RI Strategic Reporting Azure Infrastructure Release Pipeline
	myBeazley PCG AUS	PCGAusEvents v1 PCGAusEventsAltamont v1 Documents Aus Events Aus Events Altamont			
	myBeazley PCG UK	PCGEvents v1 PCGEvents v1 Documents Events Events Matrices			
	myBeazley PCG US	PCGUSEvents v1 PCGUSEvents v1 Documents US Events			
	myBeazley TV&Film	Package v1 Package v1 Documents Package v1 Schemes			
	myBeazley Marine	MarinePC v1 MarinePC v1 Documents MarinePC v1 Schemes			

Proposed Component(s)

The following list of components will need to be managed centrally to ensure consistency in labelling and reflect any changes:

MyBeazley US/UK/ROW		
PLATFORM	**PRODUCT**	**COMPONENT**
Beazley Rating	myRate	MLPackage Product v1 MLPackage Product v1 Schemes Allegiance BPI Extractor_Rulebook BI
	Safeguard & DWP	SafeGuardDWP Documents v1 SafeGuardDWP Product v1 SafeGuardDWP v1 Schemes
	Common components	Azure Infrastructure Release Pipeline

Defect Attributes by Status

M	Mandatory
O	Optional
P	Protected
H	Hidden
D	Default

Field	New	Open	Re-Open	Deferred	Duplicate	Rejected	On Hold	Fixed	ReadyToTest	InTest	Closed
Platform	M	O	P	H	H	H	P	P	P	P	H
Product	M	O	P	H	H	H	P	P	P	P	H
Product (BZD)	O	O	P	H	H	H	P	P	P	P	H
Project	D	H	H	H	H	H	H	H	H	H	H
Component	O	O	P	H	H	H	P	M	P	P	H
Sub-Components	O	O	P	H	H	H	P	P	P	P	H
Build Version	M	O	H	H	H	H	H	O	P	P	P
Summary	M	O	O	P	P	P	P	P	P	P	P
Environment	M	O	O	H	H	H	H	P	P	P	P
Description	M	O	P	P	P	P	P	P	P	P	P
Expected Results	M	O	P	P	P	P	P	P	P	P	P
Actual Results	M	P	O	P	P	P	P	P	P	O	P
Test Data	O	O	O	P	P	P	P	P	P	O	P
Defect Priority	M	O	O	P	P	P	P	P	P	P	P
Defect Severity	M	O	O	P	P	P	P	P	P	P	P
Reporter	D	P	P	P	P	P	P	P	P	P	P
Assignee	O	M	M	H	H	H	M	M	M	M	P
Test ID	M	O	O	H	H	H	H	P	P	P	P
Linked Issues	O	O	O	D	D	D	D	O	O	O	P
Label	O	O	O	O	O	O	O	O	O	O	O
Sprint	O	O	P	P	P	P	P	O	P	P	P
Attachment	O	O	O	O	O	O	O	O	O	O	O
Fix Version	O	O	P	P	P	P	P	M	P	P	P
Root Cause	H	H	H	H	H	H	H	M	P	P	P
Root Cause Description	H	H	H	H	H	H	H	M	P	P	P
Reason	H	H	H	M	M	M	M	P	P	P	P
Defect Type	O	O	O	P	P	P	P	P	P	O	P
Team	O	O	O	P	P	P	P	P	P	P	P
EPIC LINK	M	O	P	P	P	P	P	P	P	P	P

Clarifications

- The preferred route to creating a defect is via Xray or Zephyr test management plug-ins; all defects should be linked to a test ID and tests linked to a story to enable full traceability
- Defects identified via exploratory testing should be linked to Story/Epic
- Defects identified outside of the sprint are managed using Epic Link and Fix / Release version.
- Defect Severity and Defect Priority fields will be entered by Testers while logging the defect
- Once a defect is set to fixed the Root Cause and Root Cause Description should be set by the individual who has fixed the defect.
- To support access to JIRA by Third Party organisations, user accounts will be created so defects can be assigned to third parties and a notification is emailed to them. There are some third parties that may already have a Beazley network account or access to the Beazley network from Citrix etc. Where appropriate these third parties should continue to use this method to access JIRA.
- Test and Defect traceability will be established from OKRs > Defect(s) using our standard dashboard setup
- Environment defects will be logged via the Environment list configuration
- Production (software) related defects logged via ServiceNow will be logged in Jira in the usual way until we have a working bi-directional integration.
- “Test ID” field is used to link defect to Test - once Xray is implemented the test id will be populated automatically from test execution.
- Once triaged, defects will be assigned to the individual setup as default for the component
- Use ***Deferred*** status for a defect which cannot be resolved within the sprint and/or maybe classified as a Change Request (CR).
- Use ***On Hold*** status for a defect where a dependency needs to be met before it can be resolved or where waiting for some clarification.