Inland Revenue
Te Tari Taake

Inland Revenue

Inland Revenue

Test Strategy

Enterprise Design and Integrity – Strategic Architecture

Senior Responsible Owner:

Prepared by:
Date: 1 April 2022

Click or tap here to enter text.

About this Document

This artefact provides the Test Strategy for Inland Revenue. It encapsulates all aspects of testing to support delivery and BAU change at Inland Revenue. This artefact is a reference artefact providing Inland Revenue standards for testing, test phases, test entry/exit criteria, defect severity definitions, test reporting, test environment and security controls.

Document Control

File Name and Path	
Contact Person	
Status	
Version	
BT ID #	

Document Review History

No	Date	Change Description	Contact
v0.01	01/04/2022	Initial document	
v.08	17/05/2022	Updated following Security and Design Authority Feedback	
V9	02/06/2022	Approved at Design Authority	

Document Signoff

Formal Review Area	Name	Signature	Date
Responsible person *Enterprise Design and Integrity – Strategic Architecture | Domain Lead Testing*			
Accountable Person *Enterprise Design and Integrity – Strategic Architecture | Enterprise Leader*			

Formal Review Area	Name	Signature	Date
The following people have supported the development of this document: *CCS | Domain Lead* *CCS | START Delivery Manager* *CCS START | Domain Specialist* *CCS Digital | Test Manager* *CCS | Domain Lead* *Business Transformation - Test Manager* *Enterprise Services | Test Manager* *Enterprise Services | Technical Specialist*			
The following people and groups have been consulted: *Customer Compliance Services - Planning Design and Delivery | Enterprise Leader* *Enterprise Services | Enterprise Leader* *Information and Intelligence Services | Intelligence Leader* *Chief Information Security Officer* *Strategic Portfolio Stewardship | Enterprise Leader*			

Contents

1 Executive Summary

This artefact provides the Test Strategy for Inland Revenue.   It encapsulates all aspects of testing to support delivery and BAU change at Inland Revenue providing Inland Revenue standards for testing, test phases, test entry/exit criteria, defect severity definitions, test reporting, test environment and security controls. This artefact is a reference artefact, intended to reduce waste by applying the concept of writing once to be used for all testing in the organisation.

This artefact aligns;

- Future Operating Model
- Enterprise Design and Integrity Blueprint
- IR's Future Testing Services Model – approved Strategic Governance Board 3 June 2021

This artefact covers the Inland Revenue Organisation

- Customer Compliance Services
- Enterprise Services
- Information and Intelligence Services
- Enterprise Design and integrity

The Test Strategy covers the following:

- Test Organisation
- Test Phases
- Test Entry and Exit Criteria
- Defect Severity Definitions and Process
- Test Deliverables and associated RACI
- Test Environments
- Security and Data Compliance

The Test Strategy is supported by the Inland Revenue Testing Patterns standards which sets out the application of the Test Strategy across the Inland Revenue Business and Technology Landscape.

What’s new

Organisational alignment

- Reflection of the Future Operating Model in roles and responsibilities in “Test Organisation”
- Testing services model updated in alignment with IR’s Future State Testing Model – SGB Paper, this is reflected in the “Testing Services Model”
- Introduction of the ‘Initiative’ context in alignment with Strategic Portfolio Stewardship, this is reflected in the “Testing Programme”

Evolving Testing

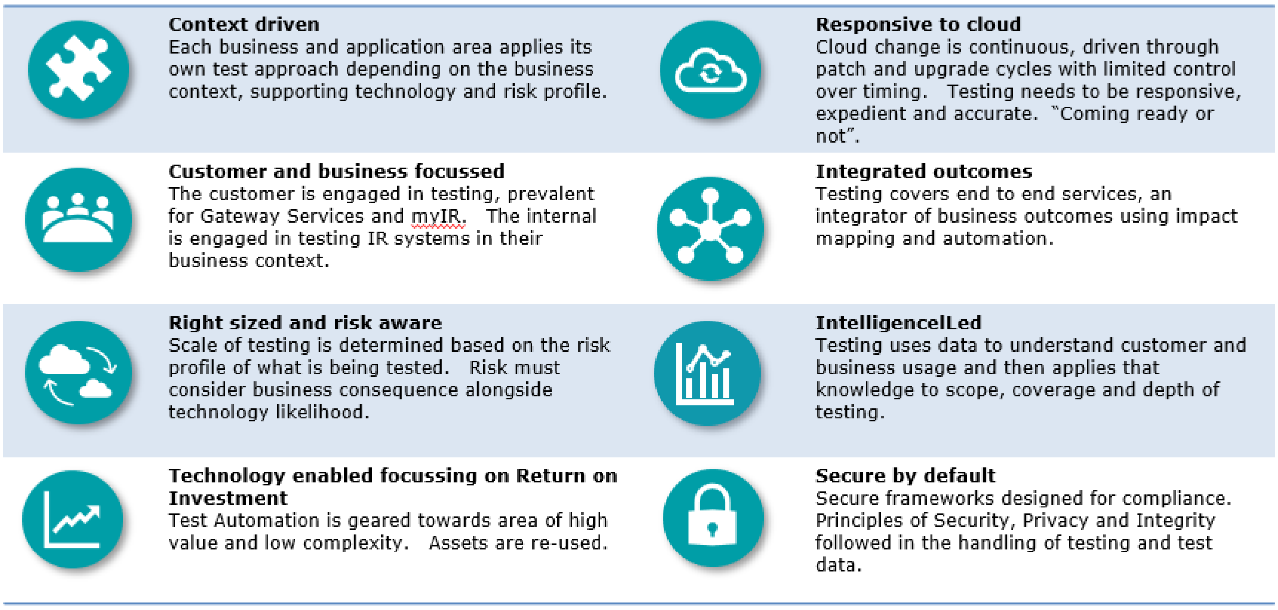
- Accessibility testing has been updated to reflect recent learnings, a pattern is provided to address accessibility
- Testing Environments now reflect the post Transformation context – i.e. rationalised Test Environments

What’s next

- Digital Inclusivity is an emerging theme in our digital landscape, there is an open question ‘what is the role of testing in Digital Inclusivity’. We have seen testing evolution in the areas of Customer and Accessibility. Digital Inclusivity is a challenge for the whole of society, not bound by IR or testing as a capability.
- Extension to Cross government Testing. The Commissioner of Inland Revenue is designated as the System Leader for Service Transformation. System leadership for service transformation is about ensuring government services are joined-up and organised around the needs of New Zealanders. The intent of service transformation is to improve the way government services are delivered by making sure they are accessible, efficient, and integrated.

Test Principles

The test principles provide key guidance to the programme across all areas of activity from design, communication and engagement, procurement, and operations.



Context driven Each business and application area applies its own test approach depending on the business context, supporting technology and risk profile.	**Responsive to cloud** Cloud change is continuous, driven through patch and upgrade cycles with limited control over timing. Testing needs to be responsive, expedient and accurate. "Coming ready or not".
Customer and business focussed The customer is engaged in testing, prevalent for Gateway Services and myIR. The internal is engaged in testing IR systems in their business context.	**Integrated outcomes** Testing covers end to end services, an integrator of business outcomes using impact mapping and automation.
Right sized and risk aware Scale of testing is determined based on the risk profile of what is being tested. Risk must consider business consequence alongside technology likelihood.	**IntelligencelLed** Testing uses data to understand customer and business usage and then applies that knowledge to scope, coverage and depth of testing.
Technology enabled focussing on Return on Investment Test Automation is geared towards area of high value and low complexity. Assets are re-used.	**Secure by default** Secure frameworks designed for compliance. Principles of Security, Privacy and Integrity followed in the handling of testing and test data.

The Test Strategy provides:

- ***Consistency***, common criteria across Inland Revenue e.g. Test Entry/Test Exit, Defect criteria i.e. using the same language
- ***Compliance and Consultation,*** consistent application of RACI ensuring that consultation and approval meets Inland Revenue standards
- ***Defined deliverables and standards*** that will be adhered to and endorsed. Recognising different ways of working across IR, emphasis is on required outcomes as opposed to enforced methodology.
- ***Re-usable documentation*** such as this artefact, allowing programmes to reference and reduce duplication
- ***Alignment to Business Services***, a common Test Pattern approach tailored to meet the outcomes of each Initiative's in-scope Business Services.
- ***Emphasis on Customer,*** ensure that customer focus is part of testing practices
- ***Test Automation,*** enabling broad scale assurance of IR services by way of automated testing running using continuous frameworks.
- ***Testing in context,*** providing approaches in testing that fit the context of the business outcomes and solutions being delivered.
- ***Tooling,*** supporting delivery and BAU using right-fit tools that fit the context of the processes and solutions being delivered.

2 Test Organisation

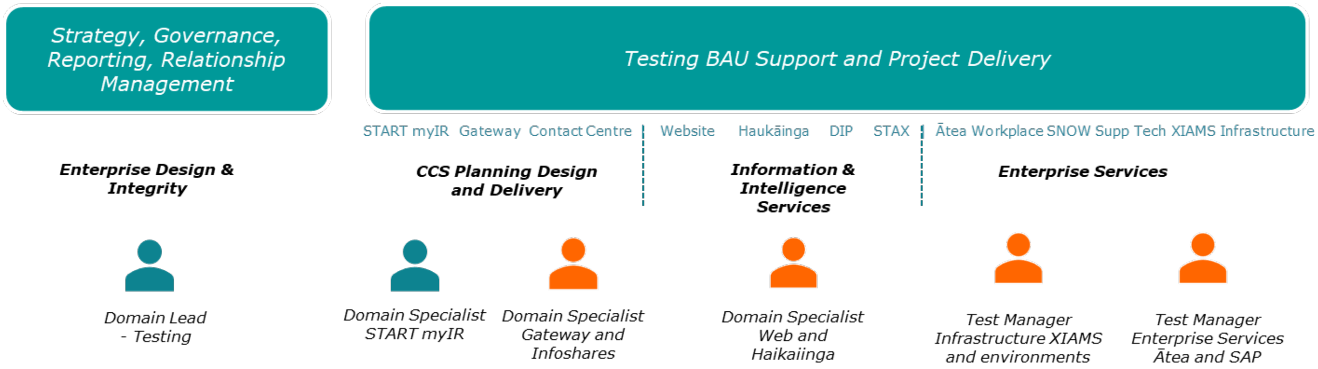

Strategy, Governance, Reporting, Relationship Management
Testing BAU Support and Project Delivery
START myIR Gateway Contact Centre
Website Haukāinga DIP STAX
Ātea Workplace SNOW Supp Tech XIAMS Infrastructure
Enterprise Design & Integrity
CCS Planning Design and Delivery
Information & Intelligence Services
Enterprise Services
Domain Lead - Testing
Domain Specialist START myIR
Domain Specialist Gateway and Infoshares
Domain Specialist Web and Haikaiinga
Test Manager Infrastructure XIAMS and environments
Test Manager Enterprise Services Ātea and SAP

Figure 1 - Test Organisation

2.1 Roles and Responsibilities

The testing organisation is established via a networked structure.

- Enterprise Design and Integrity *Domain Lead - Testing* providing for Test Strategy, Governance, Reporting and Capability. ED&I manages the strategic relationship with IR's testing partner and testing from delivery partners.
- Testing teams report into the organisation to which the testing delivery is for CCS, IIS, Enterprise Services.

2.2 Testing Services Model

CCS, II&S and Enterprise Services provisioned with a lean base testing services profile to cover BAU and a small-scale change

The model provides the ability to flex for change and scaled initiatives;

- Functional Testing and User Acceptance Testing by sourcing business resourcing on demand from Change Gateway
- Test Management, Technical Testing and Integration by way of professional testing services, IR's testing partner and technology partners as applicable.

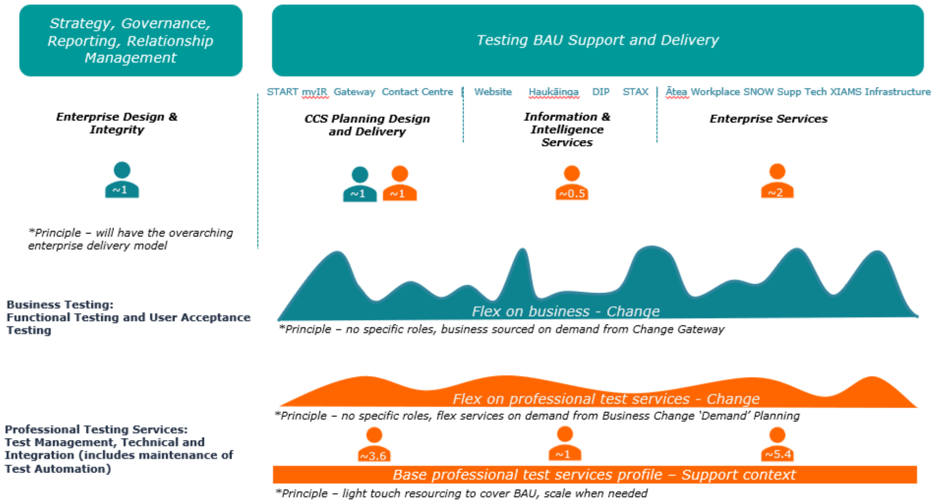

Strategy, Governance, Reporting, Relationship Management
Testing BAU Support and Delivery
START myIR Gateway Contact Centre
Website Haukāinga DIP STAX
Ātea Workplace SNOW Supp Tech XIAMS Infrastructure
Enterprise Design & Integrity
CCS Planning Design and Delivery
Information & Intelligence Services
Enterprise Services
~1
~1
~1
~0.5
~2
*Principle – will have the overarching enterprise delivery model
Business Testing: Functional Testing and User Acceptance Testing
Flex on business - Change
*Principle – no specific roles, business sourced on demand from Change Gateway
Professional Testing Services: Test Management, Technical and Integration (includes maintenance of Test Automation)
Flex on professional test services - Change
*Principle – no specific roles, flex services on demand from Business Change 'Demand' Planning
~3.6
~1
~5.4
Base professional test services profile – Support context
*Principle – light touch resourcing to cover BAU, scale when needed

Figure 2 – Testing Services Model

3 Testing Programme

3.1 Testing Scope

The testing programme is derived in the following contexts:

- The Initiative Pipelines as derived across the organisation
    - Customer Compliance Services - Planning Design and Delivery Initiative Pipeline
    - Enterprise Services Initiative Pipeline
    - Information and Intelligence Services Initiative Pipeline

- BAU Testing across the organisation
    - Customer Compliance Services - Planning Design and Delivery
        - business context supporting START, Digital Gateway and Contact Centre
    - Enterprise Services
        - business context supporting Ātea, [Information redacted] and IGA
        - technology context supporting infrastructure related change
    - Information and Intelligence Services Initiative Pipeline
        - business context supporting public website and Haukainga

The delivery pipeline is controlled within the organisational units, is prioritised by the Strategic Portfolio Stewardship by way of the Planning and Prioritisation group.

3.2 Testing Delivery Model

Testing deliverables are aligned to Initiatives, Test Delivery then occurs in the context of those initiatives.

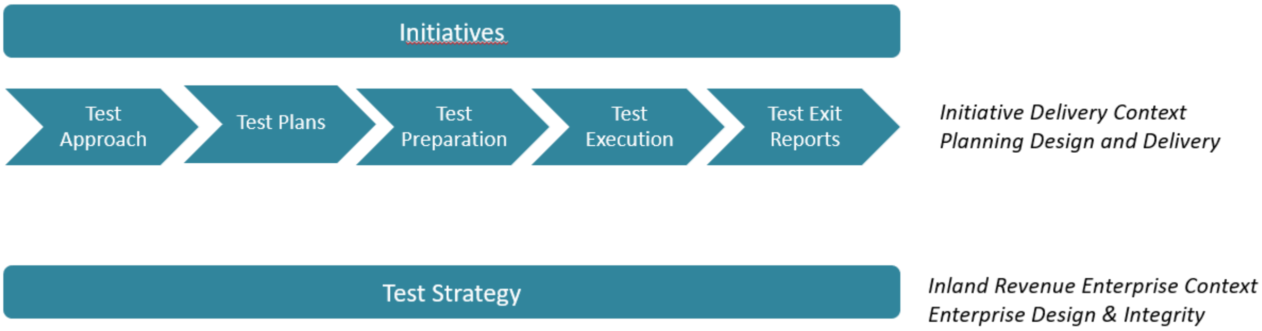

Initiatives
Test Approach
Test Plans
Test Preparation
Test Execution
Test Exit Reports
Initiative Delivery Context
Planning Design and Delivery
Test Strategy
Inland Revenue Enterprise Context
Enterprise Design & Integrity

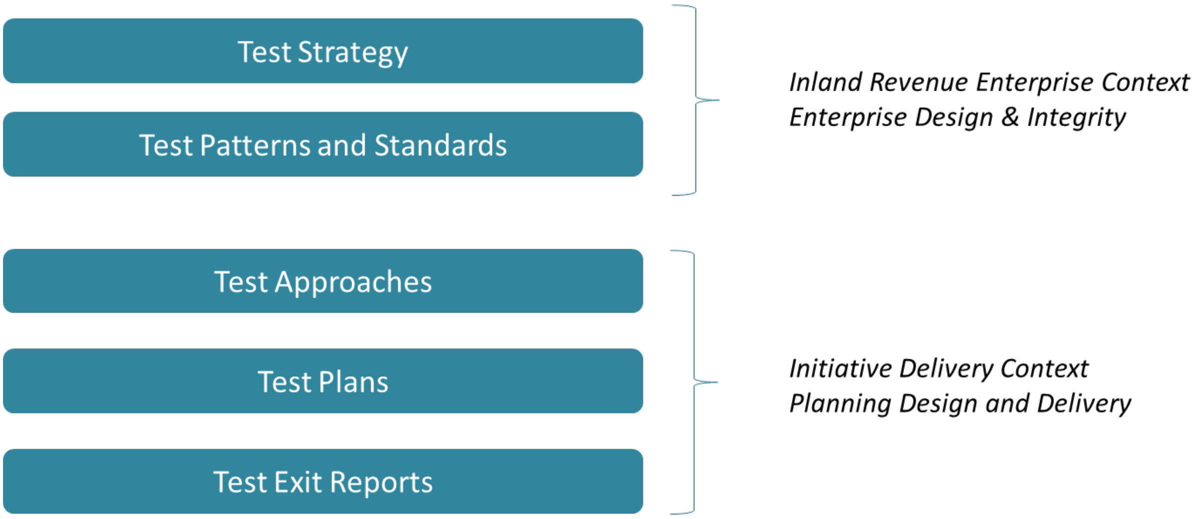
Figure 3 - Testing Delivery Model

4 Test Deliverables

4.1 Test Deliverables Hierarchy

The testing deliverables hierarchy at Inland Revenue is implemented so that detailed content of phase definitions, entry/exit criteria, RACI and severity definition is drafted once for Inland Revenue and re-used across all delivery initiatives.

- *Inland Revenue Enterprise Context* – Enterprise Design and Integrity
  - There is a single Test Strategy for Inland Revenue which defines the standards for Test Phase Definitions, Entry/Exit Criteria, Test Capability and Defect Severity Definition
  - The Test Patterns and Standards sets out the application of the Test Strategy across the Inland Revenue Business and Technology Landscape.
- *Initiative Delivery Context* - Planning Design and Delivery
  - Test Approaches provide an overarching view of test delivery on initiatives, test phases being applied, roles, responsibilities, environments and schedule.
  - Test Plans provide detailed insight into delivery of a Test Phase on and initiative, in many cases the detail a test plan provides is baked into working practice meaning an explicit artefact is not required. Examples being Test Plan as part of Solution Blueprint for Gateway Services, or Test Plan as part of Business Function Definition summary.
  - Test Exit reports provide a measure of testing conducted against defined Test Exit Criteria


Test Strategy
Test Patterns and Standards
Inland Revenue Enterprise Context
Enterprise Design & Integrity
Test Approaches
Test Plans
Test Exit Reports
Initiative Delivery Context
Planning Design and Delivery

4.2 Test Deliverables RACI and Approval

All deliverables must be formally endorsed and approved in accordance with RACI. All testing deliverables are outlined in work plans and follow contractual requirements. All deliverable sign offs must be stored in a controlled repository.

The following RACI model applies to Test deliverables.

- Consulted parties must ‘endorse’ the deliverable
- Accountable parties must ‘approve’ the deliverable.

Deliverables	Project Manager / Initiative Lead	Business Representative (Domain Lead)	Service Integration Domain Manager (SIDM)	Domain Specialist / Test Manager	Test team	Solutions Architect	Domain Lead – Testing (ED&I)	Chief Information Security Officer Representative
Test Strategy		C	C	C		C	R/A	
Test Approach	A	C	C	R		C	C	C
Test Plan	A	C	C	R		C	C	C
Test Scripts and Expected Results		C		A	R	I		
Requirements Traceability Matrix		C		A	R	I		
Test Status Report	I	I	I	R		I	I	I
Test Exit Report	A	C	C	R		C	C	C

4.3 Test Deliverables

This section provides the deliverables that are required from testing at Inland Revenue. The Initiative or project deliverables register, and the test plan will provide the deliverables as they will be applied in the context of the Initiative or project.

4.3.1 Test Approach

Typically a one-page document or slidepack that depicts the scope and techniques for an initiative or project.

Deliverable	Description	Objectives
Test Approach	A test approach considers the following elements: • test phases • test environments • roles and responsibilities • deliverables • test traceability • architecture picture with test phases • test schedule • a short list of test deliverables • a risk assessment • any techniques being applied • any test automation	• To communicate test approach

4.3.2 Test Plan

Test Plans will be created per Initiative or Test Phase depending on the objectives.

- The project team may choose to embed core material from a Test Plan into Test Approach and therefore negate the need for Test Plan.
- The project team may choose to embed core material from a Test Plan into project artefacts (such as Solution Blueprints).
- The Test Manager/Initiative lead may choose to create a single test plan containing all test phases or break it into multiple test plans for each phase.

This deliverable is used to define and communicate a detailed test approach for each test phase such as Unit Test, System Test, Integration Test, Performance Test, Business Test etc. The Test Plan explains the objectives and scope of the test, resources, key dates.

The following is a high level of what will be included in a Test Plan:

Deliverable	Description	Objectives
Test Plan	This test plan covers all test planning for a Business Service or test phase within a Initiative.	• Test scope • Test phases • Test approach • Testing roles and responsibilities • Traceability—reference the artefacts that traceability will be applied to • Environment requirements

Deliverable	Description	Objectives
		• Risks and dependencies • Test tools • Hand-over to support approach.

4.4 Test Scenarios / Scripts and Expected Results

Testing by defining test scripts is activity that uses scenarios: stories to help the tester work through a complex problem or test system. Scripts can be further broken down into sub-scripts as required, until detailed, testable scenarios and expected results can be determined.

Test scripts define the tests to be performed and should match the test phases they relate to and the requirements and designs they intend to prove:

Deliverable	Description	Objectives
Test scripts and expected results.	Test scripts and expected results for the Initiative or project test phase.	• To provide test scripts for the scope under test • To provide clear expected results • To provide a basis for shared understanding of the detail of what is being tested with all stakeholders.

4.5 Requirements Traceability Matrix

The Requirements Traceability Matrix (RTM) maps each test to the requirements for a specific test phase

The format for an RTM may be in a tool, spread sheet, mind map or another agreed format.

Deliverable	Description	Objectives
Requirements Traceability Matrix	A matrix that organises and traces tests to requirements.	• To provide clear traceability of test to requirements/user stories. • To provide a complete view of how each requirement is covered.

4.6 Test Status Report

A weekly test progress report for testing compliant with Inland Revenue reporting standards.

Deliverable	Description	Objectives
Weekly Test Progress Report	A test progress report that includes: - Risks - Issues - Deliverables	• To provide clear reporting on status and project controls.

Deliverable	Description	Objectives
	- Test Status - Defect Status (if applicable).	

4.7 Test Exit Report

The Test Exit report provides a quantitative view of the testing results. Upon successful completion of the test execution, the test phase is formally closed. This is achieved through a Test Exit Report. The Initiative or project may aggregate to a single test exit report or create test exit reports for each phase.

The Test Exit Report documents the testing conducted at a summary level, records any variance from the test plan and assesses the testing conducted against exit criteria. Any residual defects, risks and issues are documented.

A Initiative test manager may choose to create a single test exit report containing all test phases or break it into multiple test exit reports for each phase. Strong preference for a single test exit report for a release.

Deliverable	Description	Objectives
Test Exit Report	The test exit report covers test completion for a release or test phases as applicable	• an evaluation of test completion against exit criteria • test status summary • defect status summary • variance to test plan

4.8 Testing traceability

A clear line of sight must be demonstrated from requirement to test for every test phase. Traceability is an end to end programme construct; it covers the domains of analysis, development and testing. To achieve a complete view, it is important to address what activity has been conducted before testing and then how that is modelled into the test phases.

Testing quality is driven by several factors including "test inputs". Regardless of development or delivery methodology, test phases will align to design artefacts and deliverables.

Different levels of test artefacts help focus testing on the full set of functionalities being delivered from the granular unit level all the way up to the core business process.

In the START context testing traceability is provided to Business Functional Definition (BFDs), for Atea, testing is traced to User Stories.

5 Test Phases

This section describes the testing phases that a project may move through before being released into Production. The test approach will articulate which test phases apply in the context of the scope being delivered.

The following phases of testing may apply:

- Functional Testing
  - Unit/Verification Testing
  - Business System Testing / System Testing
  - Integration Testing
  - Partnership Testing
  - Regression Testing
  - Scaled Business Simulation
  - Functional Security Testing
  - User Acceptance Testing
  - Customer Interaction Testing
  - Accessibility Testing

- Non-Functional Testing
  - Infrastructure Testing
  - Performance Testing
  - Security Testing
  - Disaster Recovery
  - Operational Acceptance Testing

5.1 Functional Testing

5.1.1 Unit and Verification Testing

5.1.1.1 Expected Outcome

Testing of the components that make up the solution to ensure they are functionally correct. Tests are traceable to design or functional definition.

5.1.1.2 Definition

Unit Testing confirms that the individual code module operates in the correct manner, as an isolated unit and is performed at the completion of the development of each code module. It proves that the linkage of build components produces the correct outcomes. Unit testing will be conducted on all components of the solution by the team coding the program or interface. Where applicable the use of test stubs and drivers will simulate upstream and downstream functionality of the component being tested.

START context

Development Task Verification is an activity under the FAST Method that has project business representatives directly engaging with the development team. Verification is an iterative process where business staff confirm the configuration completed by the developers meets their needs and expectations. Business staff provides feedback on the functionality within the system and developers continue to make changes. Changes are made and re-verified, possibly multiple times, until they are satisfied.

- Mandatory Development Task Verification.
    - Developers are responsible for initiating verification by working with the Domain Specialists to go through the solution informally
    - The standard due date to complete verification is 7 days from the development completion date, however this can be extended to an achievable date mutually agreed by the team.
    - Reporting and progress visibility through Delivery Workbench
    - Defining ’Acceptance Criteria’ for Development Tasks: FAST Leads accountable to provide a brief description of what is expected to be verified. Writing the description can be delegated to a developer.
- Identifying ’Verification Testing Only’ Development Tasks that will have no test scenarios written or executed against them.
- Creating ”No development required” Development Tasks to perform regression testing as part of Development Verification Testing.
- If there are schedule pressures during the late stages of development verification and development tasks are cycling between verification and development, by mutual agreement development tasks will be closed with residual low severity defects to allow more time for higher value activities. The residual defects must be captured as SQRs before closing the development task.
- Data for verification created by Domain Specialists, assisted by developers by providing slices used in Developer Unit Testing.
    - DES Service Testers supporting where needed by creating and submitting Gateway Services payloads

5.1.1.3 Environment

Unit Testing is performed in the Development environment, which will be logically separate to all other test environments. Verification Testing is typically performed in a test environment. Refer to the Test Environments section of this document.

5.1.2 Business System Testing / System Testing

5.1.2.1 *Expected Outcome*

System Testing focuses on testing of the requirements as documented in the relevant business functional design, business rules and business process documents. It ensures the functionality at an application level is fit for purpose.

5.1.2.2 *Definition*

Business System Testing is the core functional testing phase and development phase, it runs concurrently with integration testing. For START projects this is also called Business System Testing (BST), for non-START projects, this is typically called System Testing.

Business System Testing / System covers:

- Functional Testing—confirms the system is operating according to defined business needs (by way of traceability to a functional artefact—see Traceability section) using both converted data and native data. Includes alternate condition and exception scenarios.
- Testing with Converted Data—confirms the system is operating according to defined business needs with converted data. START System Testing also includes functional testing of converted data. Testers utilise test scenarios, created by the project business representatives working alongside professional testers, to test and confirm the system is operating according to defined business needs with converted data.
- Business Process Testing – explicit testing of the end to end business processes, these will be mapped into Business System Testing and Integration Testing.
- Regression Testing—ensuring the change has not caused any failure in existing functionality, this can be in three forms:
    - Tightly coupled, the change directly impacts a shared function, there is shared data or common code.
    - Loosely coupled, the change has downstream implications in another system (i.e. START changes for Xpress Client).
    - Not coupled, no obvious connection however there is perceived risk, there may be sub-system data that is consumed by another system.

START context

Business System Testing will use signed off business functional design documents (or equivalent) and business processes and rules to derive test scenarios and provide detail to the business scenarios used during test execution.

- Clearly defined scope guidelines guiding clearly defined scope, with the accepted residual risk ratified by business authority
- Scoping is a joint responsibility of Domain Specialist – Testing and the FAST functional lead.
- All BST requires rationale that aligns with the BST scope guidelines, BST testing will generally be required under these circumstances:
    - New core functionality
    - Custom config and code. For example:
        - Child Support: Formula Assessment Calculation engine, Child Support Income Springboard, Relationship Springboard etc

        - Changes to custom build from previous releases
    - Highly visible areas – high reputational risk, high customer impact or high staff impact. For example:
        - myIR
        - External interfaces
        - Automated Letters and External Reports
    - Functional complexity hotspots
        - High volume combined with complex variables and complex outcomes: For example: P&I calculation, Offsets
        - Deviations from existing patterns
        - Data permutation driven converted data scenarios
    - BST focus is on business process outcomes instead of fields, rules and validations
        - Fields, rules and validations primarily tested in Development Verification Testing, unless the function fits one of the categories listed above
        - Business Testers external to the project using their business knowledge to test the business processes instead of granular field level testing

- BST scenario preparation and execution
    - Domain Specialist - Testing responsible for managing test scenario writing and test execution, reporting to BTM, Channel Owner and BST Manager
    - FAST Lead scenario review mandatory. Domain Specialist - Testing also spot checking scenario quality.
    - Test scenario estimates are no longer utilised. Test preparation progress is tracked on Test Scenario Group level = How many Test Scenario Groups have their Test Scenarios written. This is overlaid with a testing schedule view.
    - Tester training approach modified to support remote testing with more emphasis on jellybean-specific functional training:
        - Teams specifying the scope and schedule of functional demos to be given to testers throughout testing
        - On-the-job learning supported by Domain Specialists and developers, based on a defined roster of Domain Specialist and developer support
    - FAST to create and refresh base slices (native and converted). FAST training testers how to augment these slices by creating targeted data for test scenarios.
    - Gateway Services Testers supporting BST Preparation and Execution where needed by creating and submitting Gateway Services payloads.
    - Project business representatives will support review and prioritisation of test scenarios.

5.1.2.3 Environment

Business System Testing is performed in the Test or Quality environment. Refer to the Test Environments section of this document.

5.1.3 Integration Testing

5.1.3.1 Expected Outcome

Integration Testing will ensure the system meets the functional requirements derived from business processes. The testing is conducted between multiple systems both internally and externally.

5.1.3.2 Definition

The purpose of Integration Testing is to test the fully integrated solution in a way that reflects business use including interfaces, both internal and external to the design, with the hardware, software and infrastructure in a functionally equivalent state to production.

Integration Testing is prevalent across Inland Revenue across multiple domains. The context is important in that it often spans multiple technologies and IR supplier vendor domains, examples:

- myIR Customer login; spans www.ird.govt.nz, XIAMS, START
- IR employee onboarding; spans Ātea, ICA, Service Now

START context

For START. integration testing focuses on Interfaces and is included within the Business System Testing and Scaled Business Simulation phases. Gateway Services teams conduct Integration Testing on external services, this can also be referenced a “Service Testing”

5.1.3.3 Environment

Integration Test must be conducted in an integrated environment. Refer to the “Test Environments” section of this document.

5.1.4 Scaled Business Simulation Testing

5.1.4.1 Expected Outcome

The purpose of Scaled Business Simulation Testing is to test the solution under production like conditions and scale. The objective of the testing to create a realistic simulation of business conditions within the solution following the major launch. The testing will cover the launch itself and significant business events following the launch.

5.1.4.2 Definition

Scaled simulations are typical in START releases however not exclusively in this context. Testing of Payroll can utilise Scaled Business Simulation in the Enterprise Support Services context.

START Context

Scaled Business Simulation testing will examine full production cycles across platforms on a full set of converted production data. Data created in START by performing real-world business scenarios will also be used during this phase of testing. Scaled Business Simulation will be the final verification that the integrated solution meets IR's defined business needs.

The high-level SBS schedule is stored and kept up-to-date in Delivery Workbench.

The details of what will be covered in each pass will be determined during the SBS test planning. The Domain Leads, Domain Specialists and BAs will collaborate with the FAST Architects and Development Leads to determine key system dates and business processes that will be tested in each pass of SBS testing. When the detailed schedule is developed for SBS, it will be added to Delivery Workbench. Some scenarios will be added to Delivery Workbench before a 'trial run' is conducted.

Each SBS pass will be preceded by mock cutover activities, including mock conversion and other tasks that will occur during deployment. Mock cutover is planned and managed by the Deployment team in the Technical Cutover Testing and Production Implementation Verification testing phases. Scaled Business Simulation will focus on IR's most important business scenarios, emulating real-world usage conditions. It will also include comprehensive testing of START to ensure it delivers the required functionality described in the BFDs. Mock financial reconciliation will be planned and conducted as part of mock cutover activities and SBS passes.

The emphasis will be on batch processing using the START job stream tool to schedule and run daily, nightly, weekly and monthly job streams as a simulation of processing over multiple months. The Integration Testing of interfaces using higher, production-like volumes will also occur. Heritage batch processing is included in SBS testing, based on a risk-based assessment: any major daily, nightly, weekly, monthly or yearly processes will be run during SBS, if deemed necessary from a risk perspective. The results of all job streams will be logged and tracked. Results will be benchmarked against predetermined requirements in order to verify that batch processing can be completed within the nightly window. If deemed important based on a risk assessment, SBS scenarios can also contain negative or boundary test scenarios.

Scaled Business Simulation Testing will feature the same security controls as those in the Production Environment.

5.1.4.3 Environment

Scaled Business Simulation Testing must be conducted in the "Quality" environment. Refer to the "Test Environments" section of this document.

5.1.5 Partnership Testing

5.1.5.1 Expected outcome

Testing ensures the solution can successfully interact with partners meeting the functional requirements.

5.1.5.2 Definition

The purpose of covers testing with external parties, typically this testing links with a planned test phases and provides a single view of IR Testing requirements to individual partners. Where relevant, Partnership testing will have specific focus on data reconciliation.

Partnership Testing aggregates testing with a Partner for any test phase and runs the overall co-ordination of that testing.

5.1.5.3 Environment

Partnership Test must be conducted in an integrated environment. This may be TEST or QUALITY or DIGITAL environments that are currently in the concept phase.

5.1.6 User Acceptance Testing

5.1.6.1 Expected Outcome

UAT ensures that the solution meets the business requirements and processes.

5.1.6.2 Definition

User Acceptance Testing is typically performed by the business to validate that the developed solution meets the business requirements and processes. User Acceptance Test scenarios are based on business processes and requirements.

There is no explicit User Acceptance Test Phase on START. The test strategy recognises that user acceptance testing for START is blended with system, integration and functional testing within Business System Testing, Integration Testing and Scaled Business Simulation.

User Acceptance Testing is prevalent across Enterprise Services and II&S.

5.1.6.3 Environment

UAT will be performed as part of BST in the “Test” or the “Quality” environment, whichever is the most suitable to conduct the scenario under test.

Refer to the “Test Environments” section of this document.

5.1.7 Regression Testing

5.1.7.1 Expected Outcome

Regression testing ensures that pre-existing functionality still operates as expected in the updated environment.

5.1.7.2 Definition

Regression Testing is conducted is conducted within Business System Test, Integration Testing, Scaled Business Simulation or Performance Test phases to ensure that pre-existing functionality still operates as expected in the updated environment. Typically the regression test scope is established by way of an impact assessment conducted by the test team in conjunction with development representatives. Some areas, such as Atea, START Gateway Services and core START Child Support formula have comprehensive test automation suites, these suites are run daily and on demand.

5.1.7.3 Environment

Final Regression Test will be performed in the "Test" or the "Quality" environment, whichever is the most suitable to conduct the scenario under test.

Refer to the "Test Environments" section of this document.

5.1.8 Functional Security Testing

5.1.8.1 Expected Outcome

Testing ensures that authorised users have access to correct features as per requirements.

5.1.8.2 Definition

Functional Security Testing ensures that only authorised users have access to features of a system, and that the users have access to the functionality they need to perform their assigned duties and no more. This testing includes testing user access delegation functions. User access rights creation, updates and deletion will also be tested.

5.1.8.3 Environment

Functional Security Test will be performed in the "Test" or the "Quality" environment, whichever is the most suitable to conduct the scenario under test.

Refer to the "Test Environments" section of this document.

5.1.9 Customer Interaction Testing

5.1.9.1 Expected Outcome

Customer Interaction Testing (CIT) is designed to increase the voice of its customers (tax agents, individuals, businesses, third party software providers and more) in the testing process. CIT typically applies to external users of IR systems, it can apply to internal users.

5.1.9.2 Definition

Customer Interaction Testing gathers 'real life' experiences of customers at every iteration of development, allowing for the business teams, developers and technical staff to maintain constant contact with the people for whom the systems are being designed in the first place.

Typically Customer Interaction Testing is engaged in sprints to define key functional areas to expose the IR product to the identified customer segments e.g. tax agents, individuals and businesses. Feedback is collected in sprint reports on design issues, defects, observations, and customer experience. The feedback then helps identify design improvements for resolution.

Technology solution in IR environments enable scale, with the ability to engage thousands of end-users, and provided detailed management information on customer interaction. The technology mix is Qualtrics for customer survey alongside IR Digital Test Environments that are accessible to support real customer interaction.

5.1.9.3 Environment

Customer Interaction Testing will be performed in the "Test" or "Quality" environments, the "Digital" environment is the preferred environment. The environment must be accessible to our customers.

Refer to the "Test Environments" section of this document.

5.1.10 Accessibility Testing

5.1.10.1 Expected Outcome

Accessibility assessments of Inland Revenues (IR) systems supports inclusion of employees with accessibility impairments, using assistive technologies to perform their jobs to the best of their abilities.

5.1.10.2 Definition

IR is required to conform to New Zealand Government Web Usability Standards & Web Content Accessibility Guidelines (WCAG) 2.1 - Level AA

Inland Revenue takes a layered approach to Useability and Accessibility, it is worth noting that testing is possibly the least effective control for accessibility, in this area it is better for quality to be positioned up front in COTS, Cloud and development practices.

Layered approach to accessibility;

- *Vendor software compliance,* this is where the provider of the software solution provides certification or evidence of compliance in the base unconfigured software package. Examples are FAST providing a Gentax compliance statement or [Information redacted]
- *Delivery partner development standards compliance,* this is where an Inland Revenue delivery partner provides evidence of development practices that support compliance with accessibility standards. This controls the delivered configuration. Examples are FAST providing a development practice statement for START or [Information redacted] providing compliance of [Information redacted].
- *Accessibility Testing,* engaging with IR users with accessibility needs in testing processes. This approach provides a tailored and focussed view on areas of our systems that our Internal Customers are using.
- *External Accessibility Review,* review of IR systems by an external party, largely applicable in the myIR or www.ird.govt.nz context.

5.1.10.3 Environment

Accessibility Testing will be performed in the “Test” or “Quality” environments, the “Digital” environment is the preferred environment. The environment must be accessible to our customers.

Refer to the “Test Environments” section of this document.

5.2 Technical Testing

The role of the Technical Test team in non-functional testing detailed below.

5.2.1 Infrastructure Testing

5.2.1.1 Expected Outcome

Testing ensures the infrastructure is implemented as specified in the design.

5.2.1.2 Definition

The objective of Infrastructure Testing is to ensure that each delivered element of infrastructure is correctly implemented as documented in the Specification and the Detailed Design.

Infrastructure testing may extend to the following test types as defined in the test plan:

- High Availability
- Monitoring
- Networking
- Security
- Backup/Restore
- Audit
- Reporting

5.2.1.3 Environment

Infrastructure Testing will be performed in every environment where new infrastructure is provisioned. Refer to the “Test Environments” section of this document.

5.2.2 Performance Testing

5.2.2.1 Expected Outcome

Testing of the non-functional specifications to ensure the system delivered meets the performance expectations of the business.

5.2.2.2 Definition

The purpose of performance testing is to verify that the system meets the performance related "non-functional" specifications (NFR's / SLA's). Performance can include backup and restore speeds and loads (SAN and network speeds), restore of VMs, CDC (data synch between FIRST and START) and third-party interfaces. Also, the performance impact of security controls such as antivirus scanning load, performance impact of security controls.

Performance testing provides information to management about the quality and stability of the system so that implementation decisions can be made with an awareness of the potential production impacts. Performance testing also validates the production design for the infrastructure.

Performance testing may extend to the following test types as defined in test plan:

- Volume
- Load
- Stress
- Scalability testing
- Soak testing

To adequately measure Performance Testing of the applications, a set of Non-functional requirements detailing the allowable levels will be agreed with the Business. These metrics will provide a benchmark against which to measure results during test execution.

The performance testing framework uses the Enterprise Risk framework to quantify risk and assess the requirement for performance test.

5.2.2.3 Environment

Performance Testing will be performed in the "Quality" environment. To ensure realistic results these environments must represent production accurately. As a pre-requisite the test environment will have to be configured to allow performance testers and their tools access.

5.2.3 Security Vulnerability and Penetration Testing

5.2.3.1 Expected Outcome

Verifies the security controls defined in the Security Risk Assessments are implemented and functioning as expected

Systems and data are protected from unauthorised access and disclosure.

5.2.3.2 Definition

The purpose of security testing is to verify that the delivered solution conforms to the Inland Revenue security standards.

Security risk is evaluated with IT Security and scope is determined. External security testing services will be procured from a dedicated security testing company to conduct the testing.

Security engagement occurs at 3 levels:

- Review of technical solution and coding standards for site specific code
- Vulnerability testing of infrastructure relating to new or changed applications or infrastructure in scope
- Penetration testing.

Security testing aims to ensure that system functions as expected, Confidentiality, Integrity and Availability principles of Inland Revenue are preserved. It provides assurance that the Inland Revenue systems are securely implemented and that unauthorised users cannot access or harm data, applications and networks. It will also test the effectiveness of the implemented security controls around external channels to help secure Inland Revenue resources and minimise information security risk.

Security testing tests features to search for exposures that might result in:

- Unauthorized access to the underlying operating system.
- Unauthorized access to application resources.
- Unauthorized access to audit or authentication data.
- Unauthorized access to network resources.
- Denial of service attacks.
- Exploitation of Vulnerability resulting in loss of Confidentiality, Availability or Integrity of Data.
- Unauthorised access to Inland Revenue resources being undetected.
- Malicious activities by authorised users being undetected.

Penetration testing involves simulating an attack to test the system for potential vulnerabilities other testing covers validating such things as patch levels, network segregation and firewall rule sets

Penetration testing involves Black box testing and White box testing. Black box testing assumes no prior knowledge of the infrastructure to be tested, whereas the white box testing provides the testers with complete knowledge of the infrastructure to be tested.

For the Production system an assurance report must be produced in a publicly consumable format—ISAE 3402 Type 1 and Type 2 available on request.

Testing of START application access controls is a test activity in Business System Test.

5.2.3.3 Roles and Responsibilities

An independent IT security company, which is independent from the build team and other testing teams, will be engaged by IR to perform information security testing. The technical test team will develop an internal approval process to ensure that only authorised users are given access to test environments and test data.

5.2.3.4 Environment

Security Testing will be performed in the “Quality”, “DR” and “Prod” environments as applicable. Refer to the “Test Environments” section of this document.

5.2.4 Operational Acceptance Testing

Operational Acceptance Testing ensures that the procedures and personnel required to implement the new application are in place.

The Operational Acceptance Testing verifies the production environment's ability to handle the new system. Operational Acceptance Testing may extend to the following test types as defined in the test plan:

- IT Operations Service Testing verifies that the correct functionality, architecture, and procedures are defined and implemented to allow production support teams to run, maintain, and support the system.
- Database backup – From an Operational Acceptance Test perspective, you should test and validate that database have been backed up successfully without loss of data. Testing can be performed with help of developers or tools implemented to take the database backup at a specified time for the application developed.
- Database Recovery –Recovery and restoration could be performed either manually or automatically dependant on what mechanism is applied for the application developed. Data loss is the key here.
- Software Installation and Configuration– Software Installation and Configuration test should be performed to ensure the deployment package, scripts and configurations work according to the installation instructions. Deployed components to OAT shall be packaged and distributed to the available environment. Verification of installation, configuration and of major functionality should be done by execution of Smoke Tests.
- Rollback – testing that you can rollback an application to the last known working configuration in case problems occur during the deployment
- Failover – Failover testing verifies that an application proceeds as normal when a redundant component fails. Failover test could be performed for network failover, component failover, server failover etc
- Supportability – In Supportability testing we perform– Installation test (above), Rollback (above) and Monitoring. In monitoring we check handling of events generated when the system is malfunctioning. We perform verification of monitoring mechanisms for the system – particularly availability, performance and capacity being measured and reported.
- Reliability –Failover test and Recovery routines are explained above. You can also check that Recovery Routines work within the conditions specified.
- Performance – Generally performance testing is performed separately but sometimes from an Operational Acceptance Test you would perform sanity checks of the application under load
- Maintainability –testing installation routines and rollback for updates/patches for database, infrastructure and application.

5.2.4.1 Environment

Operational Acceptance Testing will be performed in the "Quality", "DR" and "Production" environments as applicable. Refer to the "Test Environments" section of this document.

6 Test Management and Reporting

6.1 Reporting

Test Status for Inland Revenue is provided by Test Managers to their respective BAU unit or Initiative lead. The same status is reported to the ED&I Domain Lead Testing for aggregated executive reporting.

The provision of a separate testing report is consistent with a 2015 NZ Public Sector Lessons Learnt; "The testing team need a clear communication to the governance body that is separate from the delivery arm of the project, this will ensure quality issues can be raised effectively."

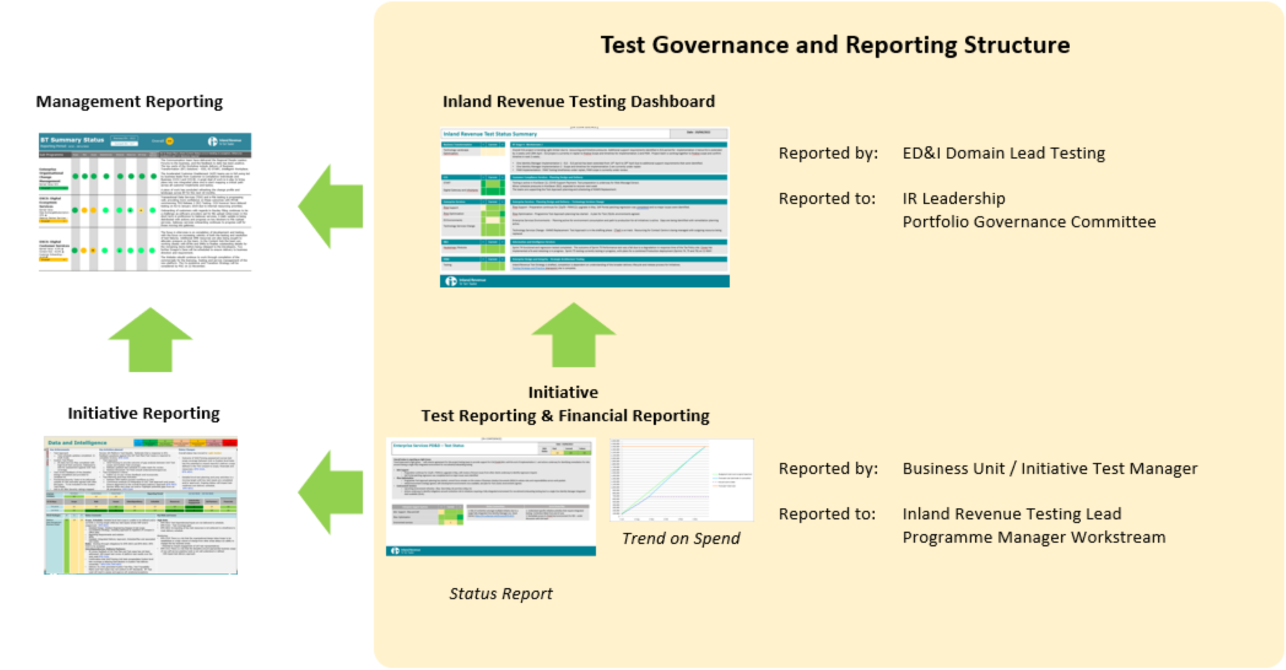

Test Governance and Reporting Structure
Management Reporting
Inland Revenue Testing Dashboard
Reported by:
ED&I Domain Lead Testing
Reported to:
IR Leadership
Portfolio Governance Committee
Initiative Reporting
Initiative
Test Reporting & Financial Reporting
Reported by:
Business Unit / Initiative Test Manager
Reported to:
Inland Revenue Testing Lead
Programme Manager Workstream
Trend on Spend
Status Report

Figure 4 Test Reporting

The following sections would be typically included in a Weekly Test Progress Report:

- Overall RAG status of Previous, Current and Next week
- RAG status for 10 keys i.e. Scope, Risks, Issues, Inter-dependency, Schedule, Resources, Finances, Stakeholder engagement and Delivery Partner
- Key Achievements in the reporting period
- Key Activities planned for next week
- Key risks and issues along with Mitigation plan/call to action
- Status of key deliverables
- Test Metrics (as applicable for IT and Business Testing)

6.2 Test Metrics

Metrics provide a means to quantify the success of the testing effort and identify areas for improvement. Metrics can illustrate whether error rates are decreasing, where errors are being introduced, whether problems are being resolved fast enough, and whether the testing process is on target. All of these can be invaluable in the management of the testing process.

At a minimum the following information should be included as part of each Test Team’s weekly status report as appropriate.

- Test Scenario progress. This includes planned and actual figures for the following:
    - Test Scenarios Completed.
- Execution Progress. This includes planned and actual figures for the following:
    - Tests Executed
    - Tests Passed
    - Tests Failed
    - Tests Planned but Unable to Be Executed
    - Tests Blocked by Defects
- Defect metrics
    - Open defects
    - Defect resolution rates

7 Test Entry and Exit Criteria

7.1 Test Entry Assessment

Prior to starting a test phase, a test readiness assessment should be carried out, this will assess the readiness to start testing and track test dependencies.

Test Entry Criteria as follows

- An approved Test Approach/Test Plan has been approved
- Sufficient approved requirements are available to construct tests
- Previous test phase is completed to the point where this test phase can start
- Tests have been written that will cover the areas that will be tested first.
- Requirements Traceability Matrix completed and reviewed (includes assessment of coverage)
- Data requirements have been identified and created within the Test environment for the scope under test.
- Test environments have been built configured to a documented state and the appropriate test data loaded.
- Configuration management processes are in place.
- A successful smoke test has been conducted.
- All resources required to execute the Test scenarios have been identified and available

It is expected that the readiness assessment is reviewed periodically on the lead up to the start of testing so that any areas that have a dependency are tracked to the test schedule. Any late running tasks can then be remediated so they do not impact the test schedule.

7.2 Test Exit Criteria (applicable to all test phases)

Test exit conditions indicate that the test phase is complete and include;

- All high and medium priority test scenarios are executed.
- All defects are documented.
- There are no outstanding severity 1 and 2 defects.
- Outstanding severity 3 and 4 defects have been reviewed and approved by business and technical representatives.
- Any workarounds are agreed by the business representatives and documented.
- For Technical Testing - Non-functional requirements have been met, or exceptions agreed upon by business and architecture after risk assessment.
- Requirements Traceability Matrix updated.
- Test Exit/Evaluation Report is approved

The test exit report must confirm the test exit criteria have been met and provide evidence to support that confirmation

8 Defect Management

Defect management is the process of logging, analysis, tracking and resolving of defects. Defects will be gathered and managed throughout all phases of testing. The Test team will report on defects and it is the responsibility of all project members to ensure defects are resolved in severity/priority order. The reporting on defects is highly visible and may be reported at any level in the organisation and potential into NZ government forums.

8.1 Defect Management Processes

Defects require a review and traction process to ensure that defects are recorded accurately, have the correct severity (according to the business) and are worked on in the correct order. The process is designed to maximise the business value of the defect process and to trend the project to meet minimal viable product, the aim is to fix the most important things first.

- *Internal Review*—During test execution, defects will be reviewed and monitored daily by the test manager (in conjunction with the relevant test analyst or business representative) and escalated as required.
- *Business Severity/Priority Review*—Business stakeholders will be briefed on defects on a regular basis (daily at peak periods). Business representatives are the authority on allocation of defect severity and priority, if there are changes required to defect severity, business consultation and agreement is mandatory. The business must provide input into the severity of every defect.
- *Defect Traction Review*—Defects will be reviewed regularly (daily at peak) with development teams to provide estimation on fix date on defects. The defect traction review ensures that defects are being worked on in severity/priority order. The Product Owner (Systems Integration and Business) from Operational Readiness and Support Initiative will be responsible for facilitating the defect review meetings with nominated members from each of the Initiative.
- *Transparency*—for any test phase at Inland Revenue, defects must be available in summary and detailed form, this includes transparency when the software is in the vendor test phases prior to the delivery to Inland Revenue.

8.2 Business Consequence and Customer Impact

Defects are often stated using technical terminology that is foreign to business and customer contexts. The Business Consequence and Customer Impact fields provide the opportunity to use plain english to describe a defect. As testing approaches exit, this information is often pivotal in determining severity and the quality level of the solution.

- *Business Consequence -* Business Consequence is a narrative that describes what the impacts are on Inland Revenues business should the defect remain unresolved.
- *Customer Impact* - Customer Impact is the narrative that describes what will happen in the customer context should the defect remain unresolved.

8.3 Defect Severity

Defect severity needs to be assessed for each defect. Severity rating will be established from the Business and Customer impacts of the defect and in collaboration with the relevant business representatives.

The following defect severity definitions are used at Inland Revenue:

Severity	Description	Examples
1- Critical	A mandatory function is affected for process or data integrity and the impact is major. No further testing is possible on that function until the Severity 1 defect is corrected. Severe business or technical problem. The system and/or integrated systems cannot be used until the defect is resolved Has a major impact on critical business activities. No workaround exists.	Defect causes system to crash. Application / solution cannot be used. Defect causes severe data corruption e.g. records cannot be resurrected. Defect causes severe loss of data. System function cannot be used. Transactions cannot be saved. Module cannot be used until the defect is resolved.
2- High	A significant function is affected and implications to the process or data integrity are major. Testing of the impacted module or system is prevented but testing in other systems/modules is possible. Serious business or technical problem. Has a major impact on business activities. Workaround exists but is cumbersome and unsustainable.	System's usability is significantly impaired. Business rules implemented incorrectly. Scenario level data corruption.
3- Medium	A function is affected but the process impact or data implications are minor e.g. a business function is affected but the implications to the process or data integrity are minor. Moderate business or technical problem. Functionality does not work as required or designed. A workaround exists.	Poor usability. Fields not defaulting to correct values. Focus and tab orders incorrect. Workflow implemented incorrectly. Errors not handled correctly. Sort orders incorrect.
4- Low	An inconvenience factors such as minor layout error, misspelling in documentation or omissions. Cosmetic defect that has minimal impact on the business. Defect causes annoyance to system users.	Spelling mistakes within application UI or internal documentation. Poor screen layout. Fields misaligned.

8.4 Approach

When a defect is found during testing it will be logged and resolved following the steps below.

1. The defect is raised.
2. The development team will investigate the defect, consulting with the test team as required and code the fix. They will update and advise the test team that the fix is ready to re-test.
3. If it is found that the defect results from missing or ambiguous requirements in specification, or the specification needs to be corrected, or businesses requirements have changed, then a change is managed. Such defects raised which require additional functionality or change in existing functionality from what is documented, will be closed once a decision on change path is approved.
4. The test team will then re-test the change and conduct regression testing if required. If the defect is resolved, then the test team will close the defect. If the defect is still not fixed then the test team will consult with the IT and/or development team, leaving the defect open.
5. If the defect is found to be raised in error, it will be closed and the reason for closure will reflect that it was raised in error.

There are two test tools FCR and Jira that will be used for defect/issue tracking see table in test tools section for details.

8.5 Transition of defects that remain open at exit

At test exit there will be times where defects remain open for further action. The test exit report will nominate the planned resolution path for a defect. Defects that may be transitioned need to be signalled to project management prior to the exit cycle along with a proposal on a resolution path.

All open defects at test exit must be documented in the test exit report along with the proposed resolution path for those defects. If the exit criteria have not been met and there are severity 2 defects at exit, then the test exit report must be supported by a contingency plan. The contingency plan is the responsibility of the Release and Deployment Manager.

If any defects require resolution in a subsequent release, then a change request is required.

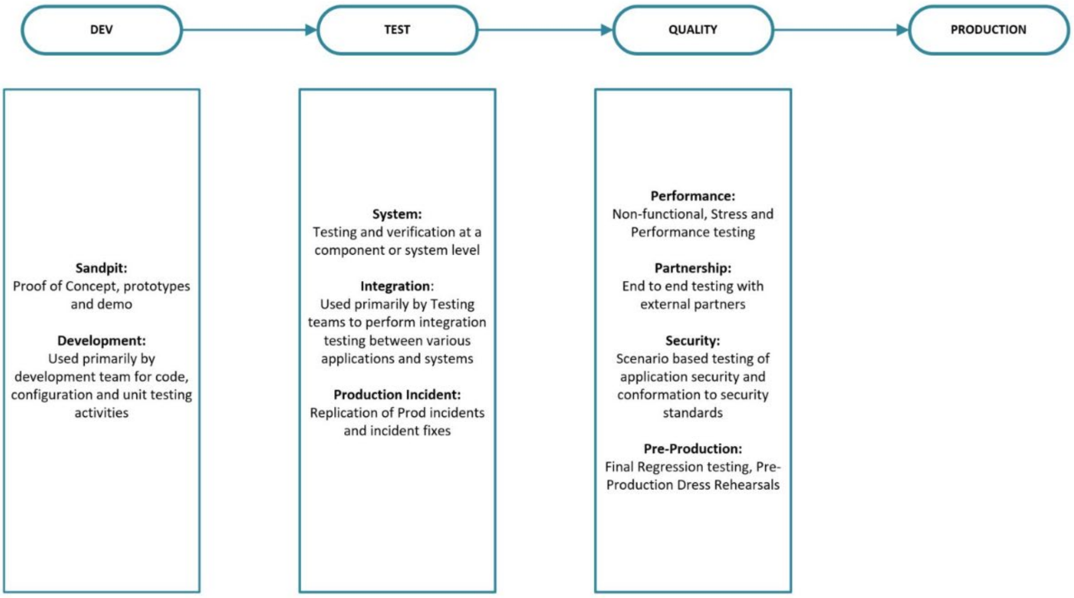
9 Test Environments

Inland Revenue Test Environment Planning is represented in the Environments Landscape Plan - MASTER.pptx

[Information redacted]

9.1 Test Phases alignment to test environments

The following diagram represents the standard alignment of test environments to test phases. The mapping is intended to be standard path implementation; it is not intended to be used to enforce constraints in test method, more a guideline. For example conducting Business System Testing in Quality is acceptable, so too is Security Testing in Test.



DEV

TEST

QUALITY

PRODUCTION

Sandpit: Proof of Concept, prototypes and demo

Development: Used primarily by development team for code, configuration and unit testing activities

System: Testing and verification at a component or system level

Integration: Used primarily by Testing teams to perform integration testing between various applications and systems

Production Incident: Replication of Prod incidents and incident fixes

Performance: Non-functional, Stress and Performance testing

Partnership: End to end testing with external partners

Security: Scenario based testing of application security and conformation to security standards

Pre-Production: Final Regression testing, Pre-Production Dress Rehearsals

9.2 Test Environment Control

Control on entry to Test Environments is provided by way of Enterprise Control.

- Enterprise Change and Release providing controls on project entry into release windows on applications and systems.
- Change Control board providing the daily control on change actions in non-production and production environments.

9.3 Configuration Management

Configuration Management is the responsibility vendor team providing the delivery services in the application or domain.

10 Test Data

This section describes the approach for test data. It defines the following:

1. The types of test data that will be used in each test environment.
2. The sources of the test data that will be used in each test environment.
3. The roles and responsibilities for the provision and management of test data.
4. The test data provisioning processes.
5. The test data management processes, including test data backup and restore.

The sources of test data include:

Source	Description
Manufactured data	Test data that is created manually. No alignment requirements.
Converted data	Data that has been extracted from Heritage applications and converted.
Reference data	Data that is copied from a reference data repository.
Production Data	Data copied or derived from Production.

The following table maps the data source to test phase.

Test Phases	Data Source
TEST Environment	Manufactured Data Converted Data Production Data – Sliced copy
QUALITY Environment	Manufactured Data Converted Data Production Data – Full copy

The data preparation for testing specific scenarios will be the accountability of the test manager for each Testing Phase.

11 Security and Data Compliance

This section covers the security and data privacy standards that must be met at Inland Revenue. The IR Testing Security and Data Usage in Tests standard is referenced here Security and Testing.

All aspects of security must be complied with including Standing Instruction 6 and 7.

11.1 Security imbedded into process

Security testing is typically back-ended in the project or Work-stream delivery lifecycle. It is important that Work-streams build security in by design and perform security validations prior to submitting systems for Certification and Accreditation. Certification and Accreditation should not be the first case where bounds or abuse cases are tested, or libraries are validated for validity.

11.2 Security controls

Security controls are to be documented in the Initiative Risk Assessment and System Security Plan.

11.3 Use of Production Data in Non-Production Environments

Data in the TEST environment potentially uses production copy data, this largely pertains to START however can occur in other applications such as Ātea. Where production data is used there are specific controls and naming standards that apply.

What is PII?

PII is any information about an identifiable individual or would allow someone to know whose information it is. In order to be identifiable, the person's name is usually attached but email address would also apply especially if someone's name is contained in the email address.

External customers - If name and email is removed then IRD number, DOB and bank account number on their own won't be PII as it's unlikely someone could be identified by that data without a name.

Internal users - Usually it's someone's name but having a userID attached to information would make it PII as you can search for a userID in the phone directory. If the employee/personnel number is attached, this is also PII but it's more difficult to identify the individual concerned as employee numbers are not widely available.

Key principles for using production data

PII (Personally Identifiable Information) is populated in sparsely in TEST environments and in the QUALITY environment, there are a number of controls and expected behaviours that users must adhere to.

All programmes and projects will adhere to the following key principles;

- Read and take note of this guidance and complete the specified training if you haven't already done so prior to accessing test environments
- Only access data that you have a justifiable business reason to do so
- If you get a security violation or are unexpectedly exposed to PII or Special Files customer make a note of what you were doing and raise a security issue.
- Do not leave your desktop unlocked whilst away from your desk
- Obfuscate, crop or delete any extracted PII where possible and label as "In Confidence"
- Only store extracted PII if you absolutely need to and store appropriately
    - Digital only to be stored in specified secured locations

  - Physical documents/media must not be left on desk and stored in a locked credenza
  - Do not leave information on printers / photocopiers unattended
- Do not create copies or share PII by email or Teams
- Dispose appropriately, extracted PII where no longer necessary
- Do not try and circumvent the process / security controls
- Do not store project or test information on your home drive.

If you are unsure about anything that you are doing please check with your manager for guidance.

The process for raise a Security Issue is to raise a Service Now incident and to tag it as "security related", if in doubt, raise the issue.

11.4 Code of Conduct

All users of PII must be familiar with the following Code of Conduct sections;

- The Accountability of Public Servants
- Decision making checklist
- Employee Obligations
- Integrity of the tax system
- Secrecy Systems Security

All users must also be familiar with the People Policies and Guidelines. More specifically the Topic 3 "Conduct and Behaviour Policy".

11.5 Inland Revenue Third Party Suppliers

The Security Remote Access Policy (which under development May 2022) will provide the procedures for managing access to IR systems and Sensitive Information for IR Vendors and Third Party suppliers.   These procedures must be adhered to.

11.6 Testing with External Partners

The Inland Revenue partner landscape includes multiple integrating parties in a number of contexts;

(i) Core Tax and Social Policy Digital Service Providers integrating with Inland Revenue's Gateway Services APIs.

(ii) Core Tax and Social Policy integration with other government agencies.

(iii) Core Tax and Social Policy integration with banking sector organisations.

(iv) Enterprise Services integration to with external organisations to support IR internal processes.

The Partnership Test plan must contain a security section and contain the following text, the wording is agreed by IR executive, please do not alter:

Inland Revenue is obliged to maintain the secrecy of any taxpayer related information. This includes information shared between Inland Revenue and business partners during system testing.

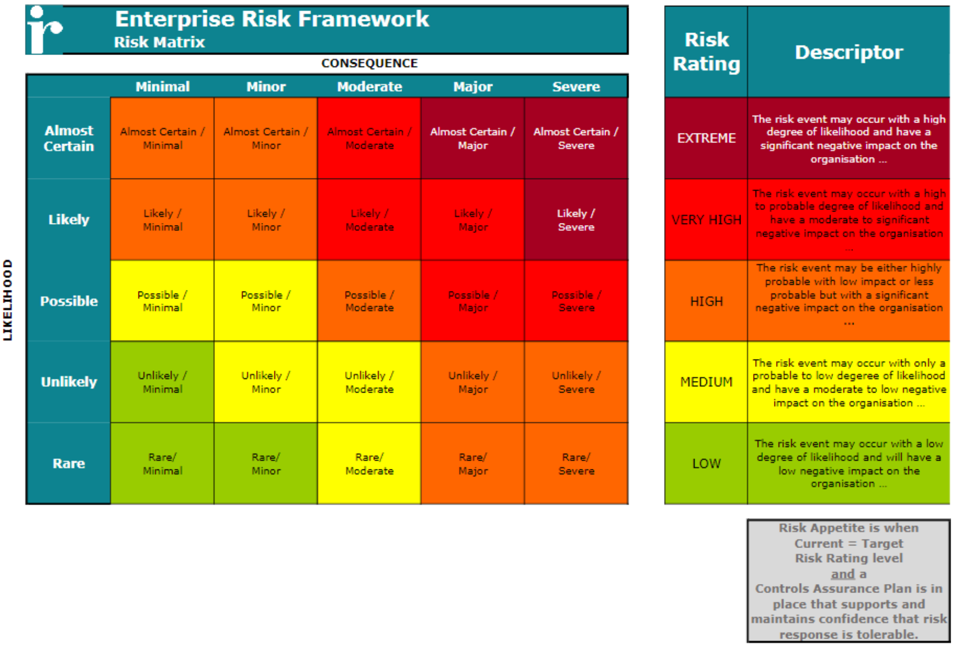
All data used in testing should be treated as if it were Production quality, with all appropriate controls in place to ensure:

- *The environment is controlled and no unauthorised individual can access the testing or staged environments,*
- *Processes are in place to ensure that source copies of data are deleted once testing is completed,*
- *The data used for testing is isolated to prevent accidental use, and,*
- *Controls are in place to prevent contamination or accidental release to other environments.*

12 Risk based testing

The testing risk model is informed by Inland Revenue's Enterprise Risk Framework. There is a balance to be achieved in the application of testing as a control on software quality and the corresponding investment by Inland Revenue in the process of testing. Typically each project or programme will utilising the risk framework to inform testing coverage. Consultation with business is essential in applying the risk framework.

- Risk Consequence Assessment - use it to assess the severity of a risk, and what is impacted by it, for example, customers, health, reputation, legislative compliance etc.
- Risk Likelihood Assessment - use it to calculate the rating of your risk based on likelihood and consequence.



Enterprise Risk Framework
Risk Matrix

CONSEQUENCE

LIKELIHOOD	Minimal	Minor	Moderate	Major	Severe
Almost Certain	Almost Certain / Minimal	Almost Certain / Minor	Almost Certain / Moderate	Almost Certain / Major	Almost Certain / Severe
Likely	Likely / Minimal	Likely / Minor	Likely / Moderate	Likely / Major	Likely / Severe
Possible	Possible / Minimal	Possible / Minor	Possible / Moderate	Possible / Major	Possible / Severe
Unlikely	Unlikely / Minimal	Unlikely / Minor	Unlikely / Moderate	Unlikely / Major	Unlikely / Severe
Rare	Rare/ Minimal	Rare/ Minor	Rare/ Moderate	Rare/ Major	Rare/ Severe

Risk Rating	Descriptor
EXTREME	The risk event may occur with a high degree of likelihood and have a significant negative impact on the organisation ...
VERY HIGH	The risk event may occur with a high to probable degree of likelihood and have a moderate to significant negative impact on the organisation ...
HIGH	The risk event may be either highly probable with low impact or less probable but with a significant negative impact on the organisation ...
MEDIUM	The risk event may occur with only a probable to low degeree of likelihood and have a moderate to low negative impact on the organisation ...
LOW	The risk event may occur with a low degree of likelihood and will have a low negative impact on the organisation ...

Risk Appetite is when Current = Target Risk Rating level and a Controls Assurance Plan is in place that supports and maintains confidence that risk response is tolerable.

An example of how the framework can be applied for applications under Performance Test Phase is as follows:

Application Name	Technical Impact (Likelihood)	Business Score (Consequence)	Inherent Risk	Derived Risk Level	Recommendation
Application 1	Rare	Severe	Medium	6	Recommend targeted Testing
Application 2	Unlikely	Moderate	Low	5	No Testing Required
Application 3	Possible	Major	High	7	Recommend Testing
Application 4	Likely	Moderate	High	7	Recommend Testing
Application 5	Almost Certain	Insignificant	Medium	6	No Testing Required

13 Test Tooling

The right fit tooling will be applied across the application and infrastructure landscape. As a principle the START toolset is largely built within the Gentax / START capability, the vendor service landscape typically use their own toolsets, IR toolsets are largely used for Integrated Testing.

Test Activities	START	Integrated Testing	Vendor Testing
Test Management and Reporting	START FCR		Within the Vendor Service
Requirements Traceability	START FCR		Within the Vendor Service
Defect Management	START FCR		Within the Vendor Service
Test Scenarios or Test Scripts	START FCR		Within the Vendor Service
Mind mapping	N/A		Within the Vendor Service
Test Automation - Continuous Integration Enablement	START Automation		Within the Vendor Service
Test Automation - API Testing	START		Within the Vendor Service
Test Automation - Contact Centre	N/A		Within the Vendor Service
Test Automation - Browser and UI	N/A		Within the Vendor Service
Customer Interaction Test – Platform	START/XIAMS Test Platform		N/A
Customer Interaction Test – web	N/A		Within the Vendor Service
Customer Interaction Testing – Customer Survey			N/A
Service Emulation (mock services)	N/A		N/A
Performance Testing			Within the Vendor Service

14 Glossary

Term	Definition
A&IM	Analytics and Information Management
BST	Business System Test – START functional, system, integration and UAT development cycle
COTS	Commercial off-the-shelf is a term used to describe the purchase of packaged solutions which are then adapted to satisfy the needs of the purchasing organisation
CIT	Customer Interaction Testing
DA	Design authority. For example, the design authority for START is the BTM.
DevOps	DevOps is a culture, movement or practice that emphasizes the collaboration and communication of both software developers and other information-technology (IT) professionals while automating the process of software delivery and infrastructure changes. It aims at establishing a culture and environment where building, testing, and releasing software, can happen rapidly, frequently, and more reliably. (Source: Wikipedia)
DRA	Deployment Readiness Assessment
DRNI	Digital Registration for New Immigrants
Enterprise Risk Model	Inland Revenue’s Enterprise Risk Framework https://irnz.sharepoint.com/sites/risk-services/SitePages/Guidelines,-frameworks-and-tools.aspx
FCR	FAST Central Repository
IKM	Information Knowledge Management
LM	Lead and Manage or LM means that the party allocated the lead and manage role will be responsible for managing delivery of the relevant deliverable or activity; This often means the party doing the deliverable or activity will not be the same party leading and managing the deliverable or activity
NPE	Non-Production Environment
NSP	New Services Platform includes the platform itself and the operations of the platform.
OAT	Operational Acceptance Testing
PGC	Portfolio Governance Committee
PIV	Production Implementation Verification
RACI	The model for governing deliverables RACI stands for: • R=Responsible • A=Accountable • S= Support • C=Consult • I=Inform

Term	Definition
RICEF	A functional decomposition method (used by Heritage Initiative) RICEF stands for: • R=Reports • I=Interfaces • C=Conversions • E=Enhancements • F=Forms.
SBS	Scaled Business Simulation Testing The purpose of Scaled Business Simulation Testing is to test the solution under production like conditions and scale
SQR	Solution Request
Shakedown Testing	A series of tests to validate that core functions are available following the delivery of environment or code. The other common term for this is Smoke Testing.
RTM	Requirements Traceability Matrix
TDA	The Technical Design Authority (TDA) for the BT Programme will provide a forum at which all technical aspects of programme design will be presented, discussed, demonstrated (where practical) endorsed and/or approved.
TDD	Test-driven development (TDD) is an advanced technique of using automated unit tests to drive the design of software and force decoupling of dependencies
UAT	User Acceptance Testing
Initiative	A programme construct within Business Transformation. A Initiative typically has a set of projects or deliverables within it.

Appendix

Key deliverables from the Business Transformation Programme.

Stage Release	Artefact
Stage 1	D180-DD - Testing Strategy and Plan - Interim Stage 1
Stage 1	D382-S1 - Core Testing Plan – Stage 1
Release 2	D257-DD - Testing Strategy and Plan - Final Stage 1
Release 2	D2210 FAST Test Plan for Stage 2 and AEOI
Release 3	D3082 - Release 3 Test Strategy
Release 3	D3064 - START Solution Test Plan - Release 3
Release 3 (applicable to all following Releases)	D3083 - Testing Patterns and Standards whole of the programme
Release 3 (applicable to all following Releases)	NZ IRD BT Program Testing Blueprint - v1.02
Release 4	D4153 Business Transformation - Release 4 Test Approach v1.2
Release 4	D4032 START Solution Test Plan - Release 4
Stage 4 WS 1 Release 1	Stage 4 Stream 1 - Release 1 - Test Approach
Stage 4 WS 1 Release 1	Stage 4 Stream 1 - Release 2 - Test Approach

RACI definitions

RACI		Description
R	Responsible	This individual is responsible for the completion of the deliverable in accordance with the agreed timeframes, budgets and quality standards. The responsible person will confirm that the deliverable meets expected quality standards before it is submitted to the accountable person for final approval.
A	Accountable	This individual, or group, has responsibility for the final approval of the deliverable. Approval is the formal confirmation that a product is complete and meets its agreed requirements. The accountable person is the only approval point for each of the deliverables.
A2	Delegated Authority	Where accountability for approval of the deliverable sits with a governance group, SRO or CEO, the authority will be given to the Programme Director to sign the sign-off memo for audit purposes.
C	Consult	These stakeholders will be consulted on the deliverable either because their input will add value, or because their buy-in is essential for ultimate implementation.

I	Inform	These stakeholders will be informed of the deliverable and receive a copy of the deliverable for information purposes.