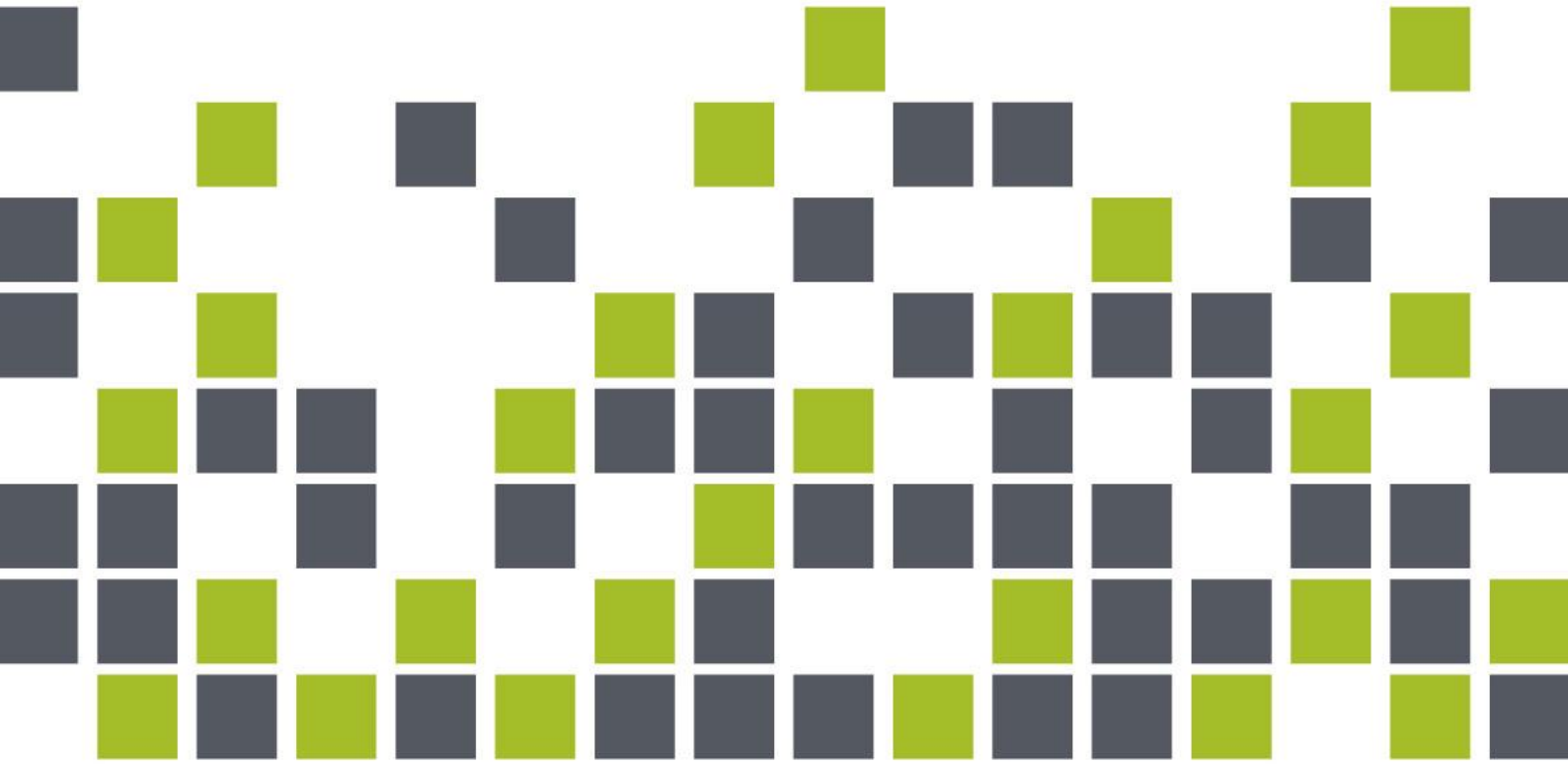
Framework for Migration of Insurance and Wealth Systems

IDIOM White Paper

IDIOM Solutions (Australia) and IDIOM Limited (New Zealand)



☎ +61 2 9968 4168 | idiomsoftware.com |
✉ Suite 16, Level 2, 357 Military Road, Mosman NSW 2088 Australia |

Table of Contents

Introduction **1**

IDIOM Solutions 1

Why Do We Need the Bridge? 2

Risk Control 4

APRA Requirement to Consider Fitness-for-Use 4

Business Knowledge and Product Cleansing 5

Glossary 5

The ISOL Approach **8**

The Major Phases for Delivery 11

Phase 1: Technical Abstraction 11

Phase 2: Business Reconciliation and Normalisation 11

Phase 3: Migration to new System 12

The Assurance Levels 12

Assurance Level One 14

Assurance Level Two 15

Assurance Level Three 15

Assurance Level Four 16

Assurance Level Five 16

Calculation Engine 16

The Control Points 17

Action Plan **18**

A Roadmap 18

Phase 1 – Parallel Abstraction 18

Phase 2 – Business Rationalisation 18

Phase 3 – Final Throw 18

Production Use 18

Product Rationalisation and Analysis 18

Selection of the Release Candidate(s) 19

Prepare for Future System 20

Map to the Future System 20
Control Point Development 20
Further Options 21
Calculation Support to the Future System 21
Direct Call 21
Service Call – Caller Supplies Data 21
Service Call – Service Acquires the Data 21
Further Benefits 22
Audit and Remediation 22
IDIOM Insurance System 23
What is the IDIOM Insurance System? 23
IDIOM Insurance System Delivery 24
IDIOM Insurance System SaaS 24

INTRODUCTION

IDIOM Solutions

IDIOM Solutions Pty Ltd. [ISOL] was created in 2017 to focus on mitigating the deficiencies and risks posed by legacy insurance/wealth systems, with specific focus on policy administration systems, but applicable across all insurance/wealth applications:

- Systems analysis, audit, and remediation
- Platform migration
- Product rationalisation
- Legacy systems renovation and extension
- Policy archive

ISOL brings together proven, proprietary tools and experienced consultants to extract legacy data and calculations, and to reconstruct these into modern, agile formats for use by any future system (see Figure 1). An additional and optional capability is Business Assurance, which is facilitated by the underlying Migration Pathway.

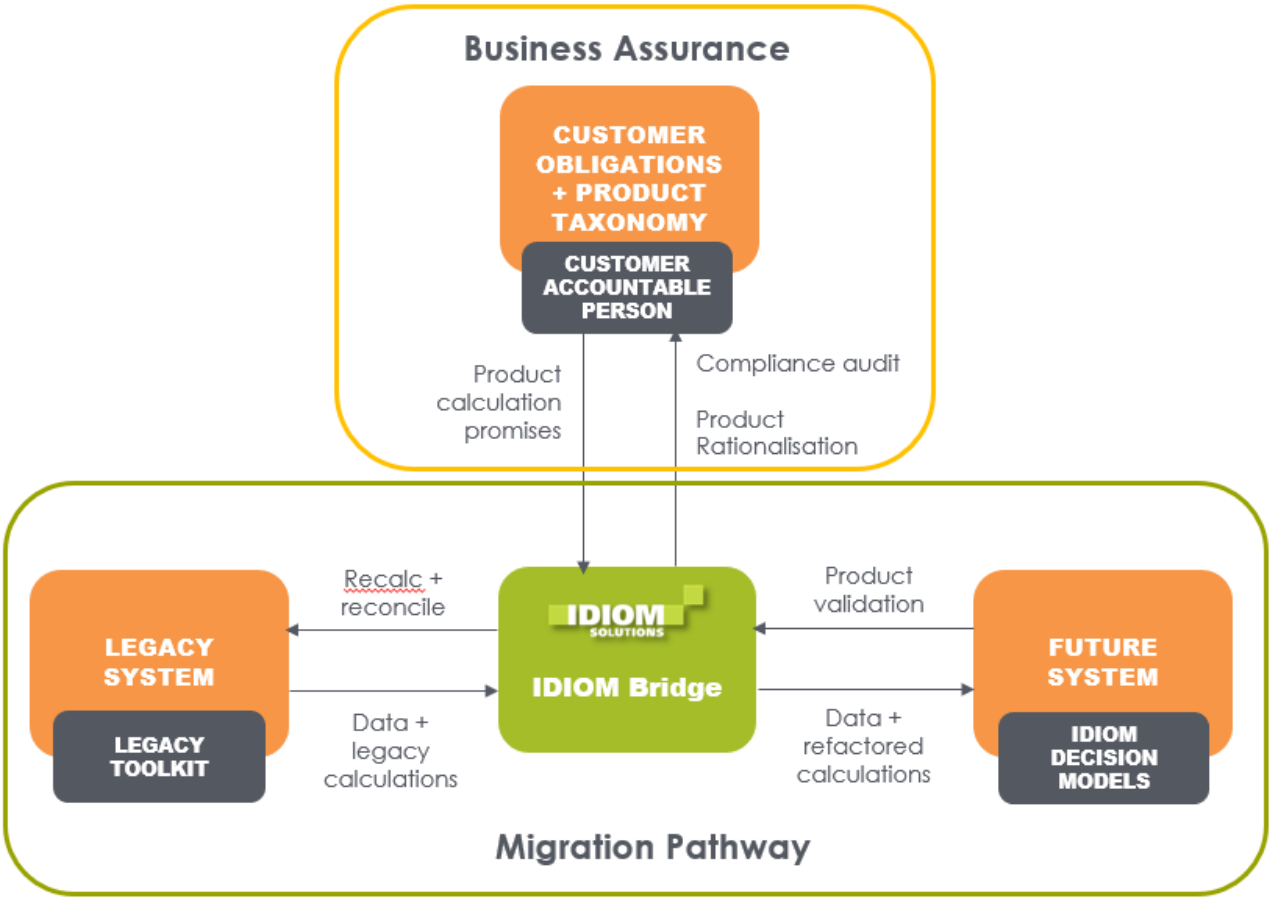


Figure 1

ISOL's legacy systems knowledge and capabilities currently spans the following systems:

- Life risk / wealth includes CLOAS, Capsil, Life70, Compass, UNISURE, Calibre, SuperB, SAS, Life400, LIFE–MVS, Talisman, Integral & Sonata
- P&C and R/I includes Polisy, COGEN, Polisy400, Huon / TIG, Sirius / PURE, Insure90, Guidewire, SAP & Duck Creek
- Health includes WHICS, HAMBS plus SAP, Amicus, TriZetto & Oracle

For the sake of clarity, ISOL is not providing generic consulting services for these systems, nor does it intend to provide 'business as usual' services in support of these systems.

The key to successful migration is the IDIOM Bridge, which provides bridging technology that can span the potentially 40year technology divide between Legacy and Future Systems.

The Bridge is a synthetic system that is comprised of a generic database and a calculation engine. It is created per client engagement to ingest and normalize the legacy system data and processes. The IDIOM Decision Manager is used to capture the calculations from participating legacy systems, including insurance product calculations (premiums, sums insured, key dates, etc.) and all associated financial and management calculations (commissions, reinsurance, etc.). The calculations are captured by Subject Matter Experts [SMEs], who are likely to be ISOL consultants for the legacy system calculations, and/or business actuaries and product owners for future system calculations. The SMEs use IDIOM's fast and effective 'drag and drop' modelling tools to document and test calculation logic, and to then automatically generate a single, all embracing 'Calculation Engine' to implement the logic.

The Calculation Engine is then used to execute the legacy calculations at scale across the full set of legacy data to ensure that the calculations have been captured correctly – this provides backwards assurance. And because the Calculation Engine comprises the full set of calculations from all source systems, functions, and products, both old and new, it can be simply injected into any new system for an assured Future System, or used to provide independent assurance for the future system.

For the sake of completeness, the IDIOM toolset also includes its own discrete application framework so that, if needed, future systems capability can be fabricated directly from the Bridge that sits center stage.

Why Do We Need the Bridge?

The fundamental problem is complexity. Most clients agree that their legacy platforms are complex. Most also consider that their target systems are complex.

To simultaneously unravel one complex system and refactor it into another unlike and also complex system, in a single logical process compounds the complexity. This complexity exists in the data and its relationships; in the calculations; and in the interplay between calculations and data. All need to be unraveled and refactored together as a single cohesive operational unit.

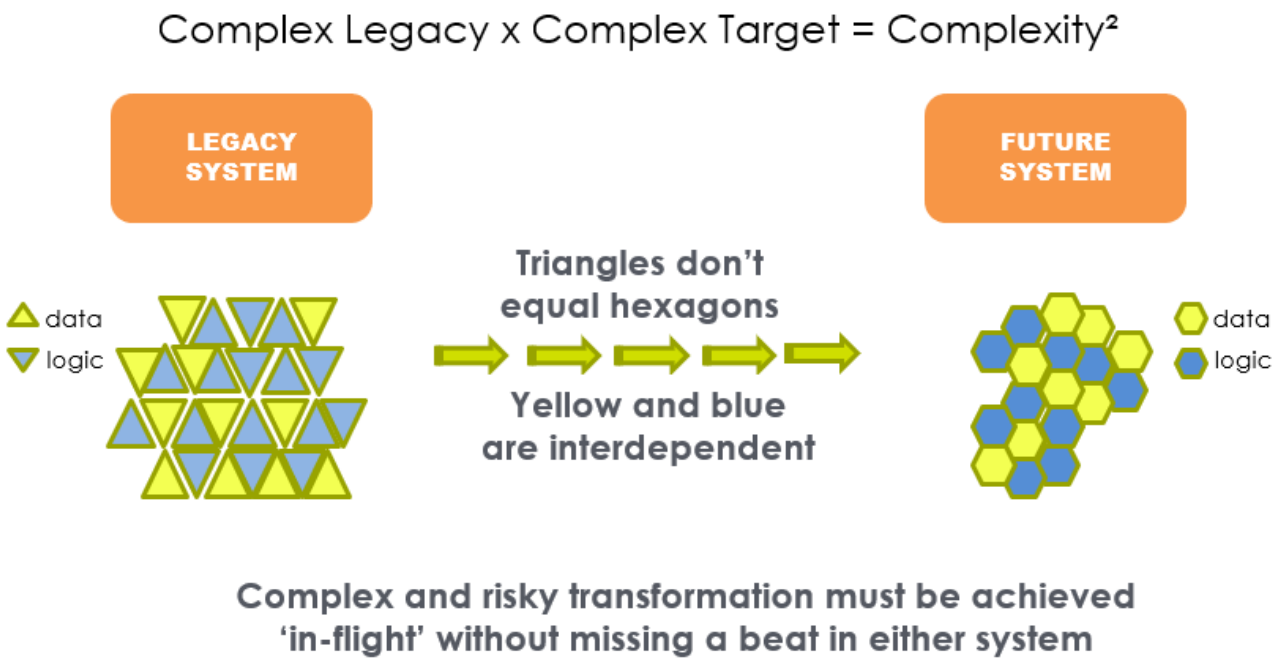


Figure 2

The ISOL approach is to break the single logical process into two simple processes, which are run back-to-back for the final throw. The ISOL tools separately address the data and the calculations, reducing each to generic intermediate forms in the Bridge.

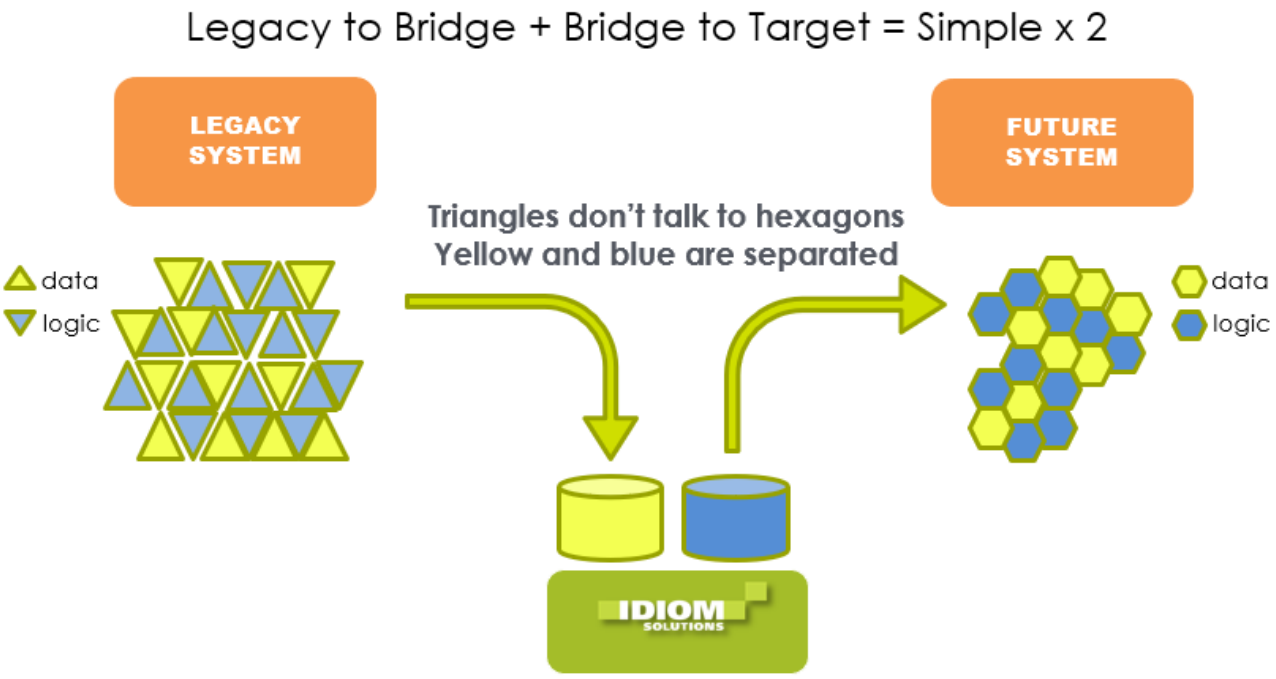


Figure 3

Migration of data and calculations from the legacy system to the Bridge mirrors only the legacy constraints, without consideration of the target system. The throw from the Bridge to the future system is similarly unconstrained by the legacy system.

The issue of complexity is resolved in the Bridge itself. With both data and calculations safely parked in a modern, industry standard generic format, we can validate, refactor, and simplify products and their calculations under the safety net of entry and exit 'control points'. These control points provide assurance that the Bridge contents match the legacy system both before and after refactoring, and they provide a benchmark for future systems use or assurance following the final throw.

Risk Control

To systematically control risk and assure end-to-end traceability, IDIOM offers a generic framework of five ***Assurance Levels*** and four ***Control Points***. This is summarised in Figure 4.

ISOL's Bridge consolidates the data and calculations. The Bridge can provide both backward assurance to the legacy system(s) and forward assurance to any new system(s).

The pivotal assurance - and also the most complex - is to be assured that data can be re-used by a new system and get the same result. This is assured by our *Assurance Level Three*, which implements *Calculations Cleansing*. This is more intricate and complex than *Data Cleansing*.

ISOL's lowest level of assurance, Level One, assures that the legacy data is correctly formatted and compliant with a data dictionary – that is to say, "named data".

ISOL's experience confirms that named data remains ambiguous unless the processes using it are fully understood; data that only complies with *Assurance Level One* will behave unpredictably once ingested by a different system. The receiving system will face significant obstacles in identifying the cause of anomalies, which requires full traceability and concurrent access to legacy source system code.

APRA Requirement to Consider Fitness-for-Use

The Australian Prudential Regulation Authority's [APRA] Prudential Practice Guide CPG235 'Managing Data Risk' places emphasis on the need for data to have 'fitness for use' before acceptance. Clause 51states:

> 51. *Data validation is the assessment of the data against business rules to determine its fitness for use prior to further processing. It constitutes a key set of controls for ensuring that data meets quality requirements*. [**APRA CPG235**]

ISOL's *Assurance Levels Two and Three* provide Validation and Calculations Cleansing to satisfy CPG235, and more generally induce early detection of issues to do with data and their related calculations. We do not believe that Data Cleansing alone satisfies CPG-235. Further information on ISOL's approach to managing CPG-235 can be found in the IDIOM

whitepaper 'Data Integrity in Financial Services' available here[1]. While APRA is an Australian regulator, the focus of CPG235 is relevant worldwide.

Business Knowledge and Product Cleansing

To be able to provide order and to maximize opportunity during a complex systems migration, ISOL recommends developing an independent view of the insurers specific customer and regulatory obligations (see the Customer Obligations and Product Taxonomy in Figure 1) as a separate activity that is done in parallel with the underlying Migration Pathway.

We call this the 'First Principles' approach because it is reconstructed from 'first principles' analysis of customer and regulatory obligations – including Product Disclosure Statements [PDS] and related customer and product documents, and the full extent of the regulatory environment. It is a conceptual view of what the insurer is obligated to provide, whether or not it does so in practice. This viewpoint provides a roadmap to guide decision making and to collate information and analysis outputs. It can also be used to guide the rationalization of insurance products, and the normalization of insurance data and processes.

Unfortunately, the history of legacy systems development sometimes means that there is a delta between the legacy systems view and the first principles view. To the extent that it exists, this delta represents regulatory risk that is becoming increasingly important as the insurance regulators increase their focus on insurer's obligations. ISOL can assist with first principles analysis, and application of first principles to the Migration Pathway, either as an active input to the migration planning and design, or as a passive audit and compliance verification of the same. It is usual to find errors, oversights, and ambiguities regarding the *first principle* obligations. In the ISOL approach, the Bridge is used to remediate and verify any corrections made, which are ultimately the responsibility of the ***Accountable Person***.

While *Data Cleansing* and *Calculations Cleansing* can help ensure that a new system works effectively, *Product Cleansing* will further ensure that it is efficient in a business context. The insurer can gain assurance via *Assurance Level* 4 that any rationalised products and related processes will continue to meet all customer obligations.

Glossary

Accountable Person means the insurer approved business person or entity that is charged with sanctioning the **Approved Formulas**. ISOL anticipates that this includes product and process owners, actuaries, tax specialists, and such like.

Alert means a record created by the IDIOM Decision Manager Workbench™ that signals a breach of a validation rule (Assurance Level Two), or that a **Controlled Outcome** does not comply with its **Business Policy** (Assurance Level Three).

Assurance Level One is satisfied when the data is valid and complies with a meta dictionary (e.g. a sum insured is a decimal amount).

[1] http://idiomsoftware.com/DOCS/Download/ce5e30f0-e6de-4b77-bfe3-766c28f395fc.pdf

Assurance Level Two is satisfied when the data is valid and also consistent with other data (e.g. a premium charge can be valid but fails consistency if the policy status – which is a separate datum – is out of force).

Assurance Level Three is satisfied when the calculation of critical **Derived Data** is replicable.

Assurance Level Four is associated with product and process rationalisation. It ensures that changes that are made to the data after Level Three to support a simplified and rationalised set of data, processes, and products, has not lost any meaning.

Assurance Level Five is the repeat application of Levels Two and Three to a new system, thereby providing forward assurance that the future system results are harmonious to the source system and/or first principles.

Bridge means the intermediate platform which contains (i) policy, claims, and/or other context data, (ii) business/product configuration and reference data, and (iii) the Formulas (i.e. the calculation logic defined within the IDIOM Decision Manager) in the form of a Calculation Engine that enables system outcomes to be recreated.

Business Policy means the business defined and approved set of methods, algorithms, and constraints that govern the calculation of **Controlled Outcomes**.

Calculation means either a) an algorithm or a Formula, or b) the result of an algorithm or Formula as the context requires. See also Decision.

Calculation Engine is an IDIOM term that refers to an IDIOM supplied executable that is used to assimilate and execute any number of Decision Models. The scale and utility of the IDIOM Calculation Engine can extend to include the entire population of calculations for most organisations.

Cleansing means:

- **Data Cleansing** means the data validation achieved via assurance levels one and two, which assures that the existing data is valid and semantically consistent.
- **Calculations Cleansing** means the outcome of assurance level three, which assures that derived data is correctly understood and reproducible.
- **Product Cleansing** means rationalising the business products, data, and processes without loss of business or customer value.

Context Data means the subject data that is supplied to a Calculation Engine to provide the real-world subject matter for any given invocation of a calculation. The context data provides the purpose of the calculation and is the proximate target for the calculation outcomes (aka 'decisions'). Context data excludes reference data, configuration data, and other passive data that are not subject to a state change controlled by the calculation.

Control Points are reconciliation points to ensure no loss, nor mismatch, of data from point of extraction to final destination in the new System. There are four recommended control points through this data journey.

Controlled Outcome is an ISOL term that refers to a derived datum that the business declares has critical importance. It should include derived data that is essential to the business mission,

and all derived outputs that are prescribed by regulation or published obligations. Reconciliation of the set of Controlled Outcomes at various stages of the migration provides the business with end to end assurance. A Controlled Outcome is assigned to/by the Accountable Person, who is the final arbiter of the **Formula** that derives the outcome value. The existence of a Controlled Outcome is enumerated in a register of Controlled Outcomes.

Decision (an IDIOM term) means the result of a Formula (aka a calculation) that is persisted because of its inherent value to the business. IDIOM formally defines a Decision as "A single definitive datum that is derived by applying business knowledge to relevant data for the purpose of positively supporting or directing the activity of the business." Because a Decision is produced by a calculation, the terms can often be used interchangeably.

Decision Model (also an IDIOM term) means "an ordered assembly of decisions that creates new and proprietary information to further the mission of the business".

Derived Data means data that has its value derived through calculation. Most critical system data is derived data, although not all derived data is critical.

First Principles means those obligations and requirements that exist independent of how the same are implemented in systems and processes. They are usually externally focused: e.g. disclosures to customers, undertakings to regulators, and legal requirements.

Formula means the specification of the logic, also known as an algorithm, that implements the **Business Policy** governing the derivation of a **Controlled Outcome**. A Formula is a reflection of and a proxy for the Business Policy that governs the outcome value. When defined in IDIOM Decision Manager, a formula is executable across the full extent of policy data within the Bridge. A Formula is 'as-built' when it represents the source system, and 'approved' when it has been sanctioned for use in a future system. If there is any variation between as-built and approved versions, then appropriate compliance documentation will be created by the Accountable Person for audit and traceability.

- **As-Built Formula** means a faithful transcription of the Formula that is derived from the existing source system.
- **Approved Formula** means a Formula that has been approved by the Accountable Person(s) and is the approved version of the Formula for all future derivations of the relevant Controlled Outcome. Only Approved Formulas are used in the Product Rationalisation process.

IDIOM Decision Manager™ is the proprietary IDIOM software that enables business Subject Matter Experts [SMEs] to build and test the Formulas that implement business policy. The IDIOM Decision Manager builds and manages Formulas, Decisions, and Decision Models, and is used to codify and test the business logic that implements business policy (for instance, product rules). Further information about the IDIOM Decision Manager™ and other IDIOM products can be found in the document 'IDIOM Transaction Engine: An Application to Manage Complex Business Entities' available here[2].

[2] http://www.idiomsoftware.com/DOCS/Download/e07606ea-5cd1-460b-bf64-569270cc03bd.pdf

System means a policy administration system, claim system, commissions system, reinsurance system, or any other system that the context requires, for insurance, wealth, or superannuation. It may be either a legacy or a future system as the context requires.

THE ISOL APPROACH

The ISOL approach to data migration is multi-dimensional:

- The primary objective is to mitigate risk by fully understanding the data and how it is used in (possibly multiple) source system's and to simplify this via a sequential three-fold cleansing:
  - data cleansing – to ensure the data is useable
  - calculations cleansing – to ensure the system behaviour in response to the data is documented
  - product cleansing – to rationalise products, data, and processes for a simpler end state in a future system(s)
- These are assured by:
  - five levels of assurance (incl. Level 5 which requires the future system to be installed)
  - four control points
- The process of execution has three distinct phases:
  - abstraction of legacy assets - which is technically driven
  - business reconciliation and normalisation – which is technically and business driven
  - the migration itself – which is relatively mechanical if the above is already completed.

The following Figure 4 (parts(a) and (b)) is an integrated picture placing these elements in context to one another and with the ISOL and IDIOM tools deployed.

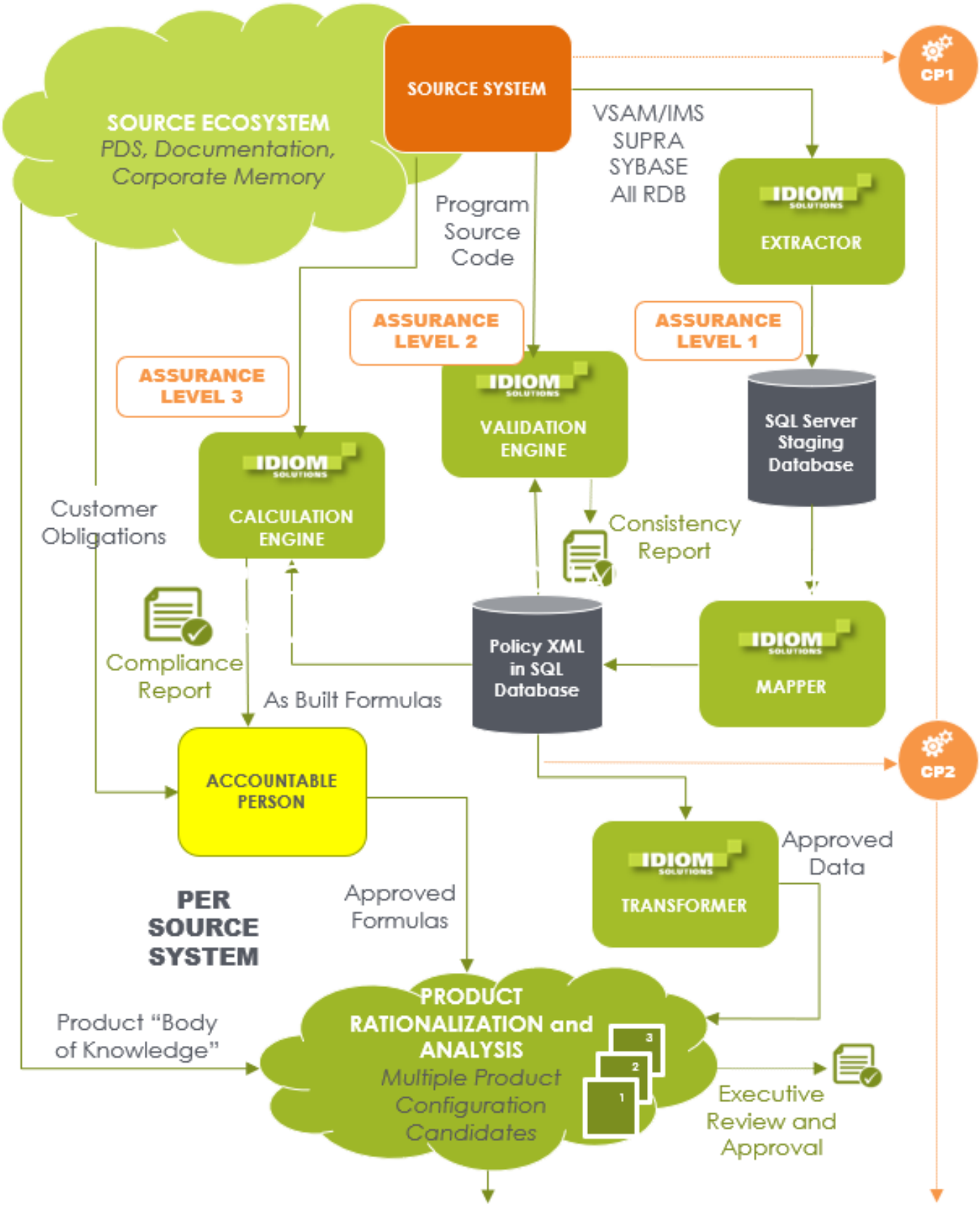

SOURCE ECOSYSTEM
PDS, Documentation,
Corporate Memory
SOURCE SYSTEM
CP1
VSAM/IMS
SUPRA
SYBASE
All RDB
EXTRACTOR
Program
Source
Code
ASSURANCE
LEVEL 2
ASSURANCE
LEVEL 1
ASSURANCE
LEVEL 3
VALIDATION
ENGINE
SQL Server
Staging
Database
CALCULATION
ENGINE
Customer
Obligations
Consistency
Report
Compliance
Report
As Built Formulas
Policy XML
in SQL
Database
MAPPER
CP2
ACCOUNTABLE
PERSON
Approved
Data
TRANSFORMER
PER
SOURCE
SYSTEM
Approved
Formulas
Product "Body
of Knowledge"
PRODUCT
RATIONALIZATION and
ANALYSIS
Multiple Product
Configuration
Candidates
Executive
Review and
Approval

Figure 4(a)

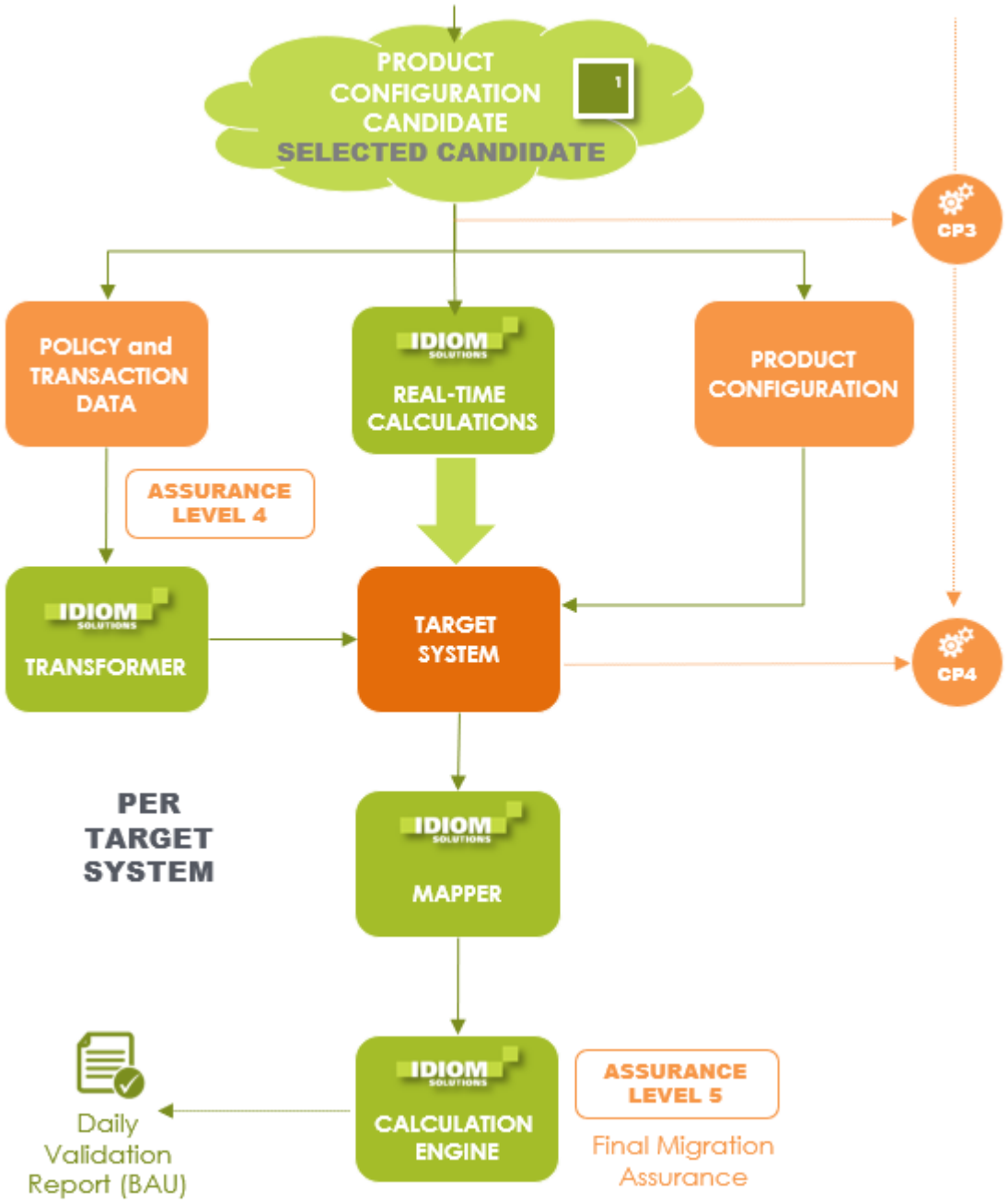

PRODUCT CONFIGURATION CANDIDATE SELECTED CANDIDATE
1
CP3
POLICY and TRANSACTION DATA
IDIOM
REAL-TIME CALCULATIONS
PRODUCT CONFIGURATION
ASSURANCE LEVEL 4
IDIOM
TRANSFORMER
TARGET SYSTEM
CP4
PER TARGET SYSTEM
IDIOM
MAPPER
Daily Validation Report (BAU)
IDIOM
CALCULATION ENGINE
ASSURANCE LEVEL 5
Final Migration Assurance

Figure 4(b)

The Major Phases for Delivery

Table 1

OBJECTIVE	ACTIVITY	ASSURANCE LEVEL
Data Cleansing	Technical Led – abstraction of assets	Levels 1 & 2 (Backwards)
Calculations Cleansing	Technical and Business	Level 3 (Backwards)
Product Cleansing	Business Led – including policy decisions to alter products and processes	Level 4
Throw	Mechanical	Level 5 (Forwards)

Phase 1: Technical Abstraction

This phase is technical and occurs for each source system. It is not a requirement of this phase for the business to clarify, correct or align business rules (which happens in phase 2).

The objective is to obtain and create an abstracted view of the source system that includes 100% of the data, and calculations as described by the Controlled Outcomes. Recalculation of the Controlled Outcomes provides assurance that the extracted data is fit for purpose.

The data is subject to assurance levels one, two and three as described in the next section before the abstracted model is considered for advancement to the next phase.

If Business Assurance is being performed, then the final stage of the Abstraction Phase is an assessment of the (recalculated) Controlled Outputs against the available ***First Principles*** product view. This assessment will conclude that the Controlled Output is compliant with all known published obligations, or it is not. A Compliance Report is created for subsequent business adjudication.

At the conclusion of the Abstraction Phase the source system is fully described and proven in the abstracted model, which can now be used without recourse to the source system.

Phase 2: Business Reconciliation and Normalisation

This phase is business driven. The primary objective is to normalise the calculations, the data, and ultimately, the products themselves into the smallest number of components that can correctly recreate all Controlled Outcomes and deliver the required customer commitments.

Before either data or products can be normalised, we must normalise the calculations, which ultimately determine the meaning of the data (the meaning of the data is the key to the data normalisation process).

Therefore, the first objective of this phase is to get a business validated and approved set of Formulas (see Glossary). Given the importance of these calculations to the subsequent processes and to the business itself, this approval must be authoritative, hence the concept of the Accountable Person (see Glossary).

The Accountable Person must formally approve every Formula that progresses into the Bridge. The Compliance Report provides input to this process. Compliant outcomes can be passed with some confidence. Non-compliant outcomes will require some compliance related action on the part of the Accountable Person – to modify the Formula, to accept that the Formula is in error, or some other corrective or mitigating action.

When the Formulas deriving the Controlled Outcomes have been approved, ISOL and insurer technical staff can normalise the calculations and then the data.

At the same time, the Accountable Person can be engaged in developing preferred product rationalisation strategies.

This process will deliver one or more candidate product taxonomies. A product taxonomy is a division of products between and within future System. While a single future system is plausible, there is often a requirement to accommodate more than one future system, in which case each future system will have its own product structure supported by Product Configurations.

The Bridge contains a set of Product Configurations and a matching set of Data (in XML form). Each candidate product taxonomy is likely to require a different set of configurations, and by extension there will be changes to both data and Formulas. Also, each candidate will require development of IDIOM decision models to do the throw from the source aligned data to the Bridge's generic format. Therefore, each candidate product structure has a real development cost and the number of candidates should be minimised (for the sake of clarity, 1 is the ideal); nonetheless, this cost is substantially less than attempting the same thing in situ in a production setting.

This phase concludes when there is exactly one candidate product taxonomy per future system, and the data and Formulas have been aligned with that taxonomy. The finalised candidate is then subject to a Control Point Check.

Phase 3: Migration to new System

The prior phases provide the necessary methods and controls to allow this phase to be relatively mechanical, albeit with strong dependency on the future system vendor. The data is again transformed to match the future system schema, and the data prepared and thrown.

Following ingestion by the new system, there should be a final Control Point Check to ensure that the throw is complete, consistent, and correct.

The Assurance Levels

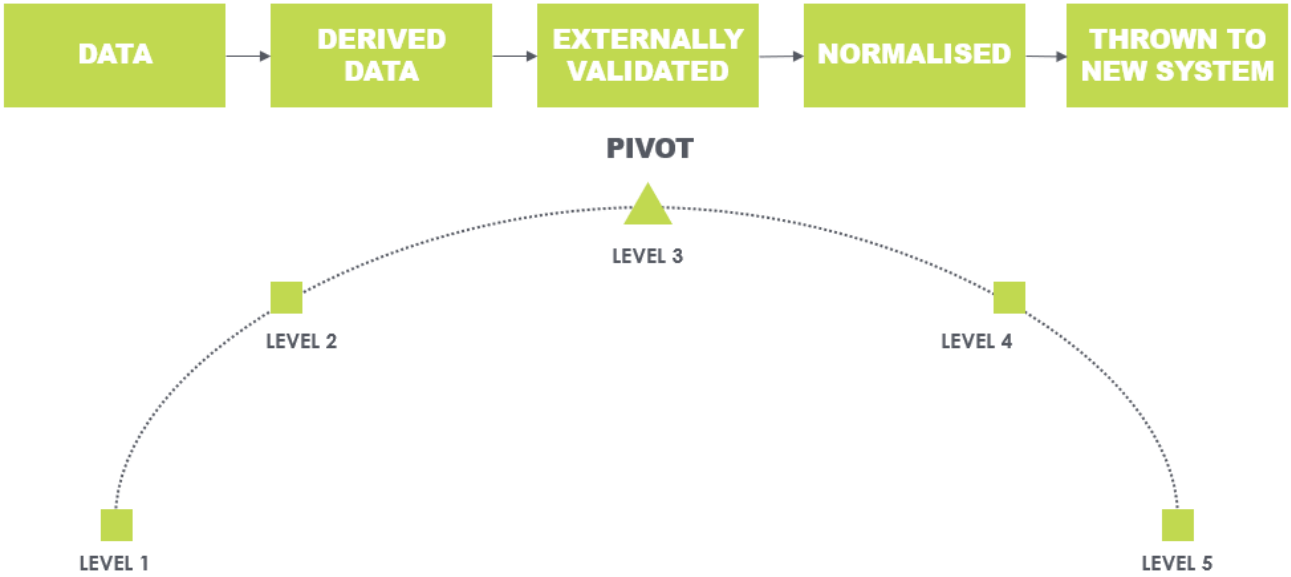
ISOL promotes 5 levels of assurance for a complete migration. The ISOL assurance levels are designed to provide a quantitative and qualitative assessment of data quality as described by each respective Assurance Level in the adjacent table [Table 2].

Table 2

LEVEL 1 – DATA FORMAT CLEANSING	LEVEL 2 – DATA CLEANSING	LEVEL 3 – CALCULATION CLEANSING	LEVEL 4 – PRODUCT CLEANSING	LEVEL 5 – MIGRATION COMPLETION
Extract and Clean ***Data is Dictionary Defined and Compliant with that Dictionary***	**Semantic Validation** ***Data is Semantically Validated against Source System Constraints***	**Assess for Compliance** ***Derived Data is Validated against Source System equivalents***	**Rationalize and Simplify** ***Rationalise Products and Calculations and Remove Source System Constraints***	**Validate in New Systems** ***Cleaned, Validated, Rationalised Product Data Revalidated in Target Systems***
Data complies with interim SQL "data definition language" definitions	Data values are tested against other data values using business rules inferred from the source system	Key values are recalculated and measured against existing system values	Data and calculations are transformed, merged, standardized, normalized, and simplified to meet future needs	New Product Configurations and matching policy data are thrown to new system(s) and validated

Of the five assurance levels, we consider Level Three the natural pivot point because this is where obligations to Customer and Regulator are made explicit.

Figure 5



Assurance Level One

Before we can build the logic that implements Assurance Levels Two and Three, it is necessary that the data must comply with the meta data definition that is being used to write the logic (e.g. it would be pointless to validate a date if the value in the date field said 'tba').

Therefore, Assurance Level One is assurance that the data complies with the meta dictionary, thus allowing progress to the further semantic levels of assurance. ISOL uses the SQL Data Definition Language [DDL] as the level one validation dictionary. That is, all transactional data[3] must be compliant with the DDL used by the ISOL 'SQL Server Staging Database'.

Note that this process may require physical changes in the data representation, including EBCDIC to ASCII, binary and hexadecimal representations to character strings, and/or changes to comply with new datatypes. This process is automated using ISOL's proprietary extraction toolset [see the 'Extractor', figure 2], providing fast and safe extraction and conversion of non-relational data.

The DDL used for Level One assurance is then re-cast into a more advanced data definition language, being the XML Schema language (.xsd[4]). During this process, additional meta information is captured, including complex (data) type information. This additional meta information enhances the DDL and is required to simplify the building and execution of rules.

The Assurance Level One compliant data is then thrown to the matching XML using the new XML Schema definition as its meta dictionary, so that the additional schema level meta data becomes usable in the context of the original transaction 'context data' (e.g. policy data).

At the same time, some data elements are normalised around the new Schema data definitions. For instance, legacy systems use many approaches to provide the equivalent of a simple Boolean (which is always true/false in XML); it can be yes/no, binary on/off, one/zero, or even blank = true, blank = false, etc. When ISOL 'throws' from the SQL to XML data representations, these anomalies are corrected to comply exclusively with the XML Boolean definition.

For the sake of clarity, the representation of data is again being intentionally changed during this process i.e. its representation is changing but not its meaning.

Because of these physical changes in the data representation (but NOT its meaning), ISOL declares a Control Point here and will revalidate the data for semantic equivalence with its source equivalent. This process is described in more detail in the section on Control Points below.

[3] Note that 'transactions' or 'transactional data' means policy data in this case. The reason that we use the generic terms transaction or transaction data is because the process is repeatable on any form of transaction data and is not explicit to policy data. In particular this may be relevant to conversion of satellite systems, e.g. CRM, Agents, etc.
[4] https://www.w3.org/XML/Schema

With the data now described by an XML Schema and compliant with that schema, we are ready to begin the level two and three assurance assessments.

Assurance Level Two

ISOL promotes the concept of semantic[5] assurance, which in this case means that the data is understood and assured in terms of its use in systems.

Semantic assurance requires that the data complies with semantic constraints. Most data are subject to constraint rules that can only be assessed by viewing the data in context using a logic engine.

For instance, a start date cannot be after the end date if both dates share the same context (e.g. a policy); each date may be technically valid (i.e. its representation, value range, and location comply with its data dictionary) when viewed individually, however the presence of the other date imposes a validity constraint that must be obeyed. The relationship between the two dates cannot be described using any existing data definition language or approach and cannot be captured using a data dictionary – the relationship is a mutual constraint on allowed values that is context dependent. Any given field may be subject to many such constraints at the same time (e.g. the start date also cannot precede the birth-date of the life insured, etc).

The context is an important concept and recognises that many values may affect the subject value being validated.

A basic set of semantic constraints are tested and assured at Assurance Level Two.

Assurance Level Three

The focus of Assurance Level Three is on the calculation of the system's critical output values, the Controlled Outcomes. This is how we obtain assurance that the data remains 'fit for purpose' in the future System, and that the future System can be reconciled back to the legacy system(s).

This is an active process, where derived data is independently re-derived. This generates a thorough understanding of the source system and traceability back to it. The alternative of passively migrating derived data values will not ensure that a future system will produce the same results ongoing as the source system did. (And for the sake of clarity and emphasis, it is this derived data that constitutes the purpose of the system.)

Assurance Level Three is a faithful recalculation of the 'as built' Formula to ensure that the source system calculation has been accurately understood and abstracted.

If desired, this can be followed by an assessment against the first principles version of the same calculation. The Assurance Level Three outcomes are a set of complete, consistent, and correct 'as built' calculations, and an assessment as to the compliance of those calculations.

[5] Semantic Defn: Relating to meaning in language or logic

Assurance Level Four

During the migration process, and particularly during the Product Rationalisation process which follows Assurance Level Three, the data will undergo various changes in its representation and relative location – it will no longer look the same. At no stage should the meaning of the data be affected by these changes unless as directed by a formally approved change in calculation method.

In order to assure this, ISOL reapplies the Level Two consistency checks and the Level Three recalculations at Level Four to provide reassurance that the Product Rationalisation process has retained outputs that match the source system. For work efficiency reasons, this assurance is required prior to providing the rationalised product hierarchy to the future System implementers.

Assurance Level Five

Finally, ISOL reapplies Level Two and Three Assurance on the new System. At this point the calculations are now being performed by the future System and should reconcile to the proven calculations contained in the Bridge.

Calculation Engine

Semantic validation can only be achieved programmatically using logic[6], which is why the IDIOM Decision Manager™ plays a central role in the ISOL data migration solution.

Building the logic requires us to work at a meta level, using the names and definitions of the data to build the semantic assurance 'idiom[7]' – this idiom is a set of rules that describe the constraints and calculations that determine the correctness of the data in the fields that are described by the dictionary.

The Assurance Level Two and Level Three assessments require development and deployment of a calculation engine to evaluate the transaction data [see the Validation and Calculation Engine(s), Figure 4]. The IDIOM Decision Manager™ is a graphical modelling tool that allows SME's to graphically model and test all required logic; it then generates high performance, native code implementations of the logic that can be executed at scale to evaluate all transactional data, both pre and post migration.

When the rules are executed, they generate new data outcomes (which IDIOM calls 'Decision's, hence IDIOM Decision Manager). These outcomes include new calculated values, and Boolean interpretations of the comparisons of values. Ultimately, when a breach of any rule is identified, the Decision Models also create an Alert for that breach. An Alert is a specific decision outcome that identifies which rule was breached by the transaction (policy) in focus.

[6] For a background on the logic used see https://en.wikipedia.org/wiki/First-order_logic
[7] A speech form or an expression of a given language that is peculiar to itself grammatically. http://www.thefreedictionary.com/idiom. It is this characteristic that gave rise to the word IDIOM in our company name.

This new 'hard data' is captured and stored in the purpose-built IDIOM Decision Manager Workbench™ [DMW], from which assurance reporting data can be derived and audited.

The DMW database also supports remediation which is further described in this document.

The Control Points

It is necessary during migration that the data is transformed in some way between various stages. For instance, we might go from EBCDIC to ASCII binary encoding in stage 1; we might also transition from primitive Assembler and/or COBOL datatypes to more modern database defined datatypes; and then we go from database defined datatypes to XML datatypes for stages 2 and 3. Then in stage 4 we propose to standardise and realign the product hierarchies which are implicit in the transactional data. And in Stage 5 we throw to the target system, requiring one last data transformation and the final validation.

The purpose of the control points is to provide an assurance of equivalence between the data contents of each system at each of the selected 'Control Point's in its migration. The control points incur an additional cost, so the location of each control point reconciliation is selected to de-risk the subsequent step, where that step is proportionately more costly. The first control point reconciliation is proposed immediately prior to Product Rationalisation, the second immediately prior to the throw to the new System, and the last following future System implementation.

The key to the control points is that the assurance is provided by an extra and quite independent process that provides similar assurance to that of double entry bookkeeping; if there is a zero delta between the migrated data and the independently sourced Control Point data at each control point then we can assert that the migration data at the control point is complete, correct, and consistent with the source.

The requirement to achieve equivalence at each control point step will require us to mirror any transformation in the representation of the data that is occurring at any subsequent step. This implies logic within the derivation of control point comparison data. Therefore, we derive the Control Point comparison data using IDIOM Decision Models.

ISOL's approach to Control Point validation is to independently (and redundantly) maintain the logic within each control point extract that is required, so that we maintain consistency with subsequent steps in the migration. When this is done correctly, the newly generated control point data will match any subsequent control point and vice versa.

The actual assessment of a match will be done by comparing each field in the schema with its match partner. Any mismatch is a fatal error that must be corrected by correcting the underlying transformation or its Control Point equivalent as required, until we have absolute and repeatable equivalence between the Control Point data and the data as it exists in the migration pathway.

ACTION PLAN

A Roadmap

In order to move forward using the ISOL approach, the following steps need to be taken.

Phase 1 – Parallel Abstraction

The Abstraction Phase can be executed independently for each source system. Allow 2-6months per system, depending on the age and complexity of the system. Systems can be done in parallel if multiple systems.

Note that there is an end-to-end dependency between the completion of Phase 1 for all systems, and the completion of Phase 2, the product rationalisation process.

Phase 3, the final throw(s), cannot be contemplated until Phase 2 is completed. By inference, the Phase 1 abstractions are initially on the critical path.

Phase 2 – Business Rationalisation

The product rationalisation phase as proposed is driven by the Accountable Person. Given the Phase 1 activities above, we expect to have a stream of issues that require Accountable Person adjudication.

An early start on Phase 2 will help ensure that the time that Phase 2 is on the critical path is kept to the minimum.

Phase 3 – Final Throw

We expect that the final throw can be prepared and executed in a timely manner when the future System is ready. The lead time for future System preparation should ensure that the final throw is not on the critical path.

Should an ISOL supplied System be required (see later in this document) then ISOL suggests a three-month preparation phase, to be followed by a 100day development sprint. The entire ISOL System provisioning should be able to conclude within 6months elapsed.

Production Use

All of the preceding steps must be able to execute in a single contiguous stream, to be repeated at will for testing purposes, culminating in the final, single-step production throw(s).

ISOL experience to date suggests that this can be usually be achieved system by system within a 48hour window per (i.e. over a weekend) without requiring disruption to business-as-usual that is attributable to the migration process itself.

Product Rationalisation and Analysis

Product rationalisation and analysis will be driven by and coordinated by insurer business leaders and the Accountable Person.

The Bridge is a sandpit where candidate product taxonomies can be developed and tested. Unwanted candidates can be discarded, while more promising candidates are further developed. One final candidate taxonomy will survive in the form of a preferred Product structure for each (if more than one) future system.

The total number of candidates is expected to be one only, however the important factor is that different product taxonomies can be contemplated without undue overhead or cost.

Each future system anticipated by the candidate taxonomy will have its own instance of a Bridge.

The data in each Bridge is a combination of product reference data [Product Configuration] and policy transaction data. The Bridge Product Configuration data will be synthesised from all of the products that are to be included in that candidate's product family and may be held (in the Bridge) as simple tables, relational tables, or XML as convenience requires.

The Bridge policy data will be held as xml in a format that is consistent with the requirements of the Product Configuration. It is a design objective (but not a technical imperative) that there be a single omnibus policy schema for all policy types.

Each candidate will be supported by a series of IDIOM Decision Models that implement the Controlled Calculations, and all transformations and reconciliations.

The primary decision model roles for each Bridge will include:

- Transformation: Decision Models will be required to throw the PolicyXML from the source derived schema into the candidate policy schema. If more than one candidate, these models will be normalised (i.e. shared elements will be reusable) to reduce development overhead. ACORD is used as a default throw format.
- The approved calculations will be adapted to use the target Product Configuration and the PolicyXML to the extent required to recreate the source system Controlled Outcomes. This recalculation will be an exact match between source and Bridge unless varied with Accountable Person approval.

Selection of the Release Candidate(s)

There is the potential to throw to multiple future systems. This could be desirable if the customisation cost for the preferred System is excessive for the strategic value of the products being thrown. A more flexible, lower cost System option may be preferred for these products, with ISOL offering its technology as a candidate.

It is highly desirable that the Product Configuration data design that is assumed by the future System is used as the starting point for the candidate (product taxonomy) selection process. The rationalisation objective would then be to reduce the number of defined products to the smallest (or otherwise optimal) set that can describe the largest set of policy instances, with priority given to the on-sale products, followed by the in-force products.

The ISOL based re-creation of the Controlled Outcomes will then empirically confirm whether or not the existing future System Product Configuration can adequately describe the candidate product set, and/or what customisation and additional features and functions

may be required. This in turn may be valuable input to finally confirming any demarcation of products for each future System.

This may be an iterative process, leading to an eventual agreed product taxonomy, which may include different Product Configuration approaches for different product hierarchies if the complete portfolio is to be allocated across more than one System.

When agreement is reached on the allocation of Products to one or more future systems, then the next task will occur for each target system.

Prepare for Future System

When the Product Structure Release Candidate(s) have been selected, further decision model development should occur for each as follows.

- Normalise the idealised calculations by an iterative process of aggregating and conditioning calculation components and adjusting both the Product Configuration and the PolicyXML as required for maximum simplicity and performance. This process must not materially change any calculated value.
- Produce a Control Point extract of the finished database and reconcile to previous control points. This reconciliation must be exact.

Map to the Future System

The process is now ready to throw to the future System. The Product Configuration is now in a tested state and is known to work in the context of the PolicyXML. Both the Product Configuration and the PolicyXML schema will now be familiar to the target system vendor and can be validated in advance. Mapping to the specific schema (DDL or xsd, including ACORD) that is used by the future system should be mechanical and low risk. By definition, any mapping conflict will be known in advance and should have been resolved through customisation of the target system.

The actual throw can then occur repeatedly as required until a cutover is made.

The final step is a further control point extraction from the future system database. This control point must run prior to acceptance and cutover to validate against the prior control points.

Control Point Development

The control point approach is an additional and discrete development effort that provides an independent confirmation that thrown values have been correctly received at key points.

Additional and independent effort is required if the process is to accord with CPG235, which requires an independent audit. This additional overhead should add between 10-20% to the overall migration development cost described in this document, depending on the intensity of assurance required.

The specifics of the approach require an extract per control point. Note that once the data is in PolicyXML form, this extract is a simple file read.

An IDIOM Decision Model is required to map each succeeding Control Point to its predecessor. This model will mirror the transformations made in the underlying forward transformations, thereby providing the independent confirmation required. The Control Point process is a closed loop, where the data transformations described in the preceding steps are re-done in the Control Point model so that the control point output should match exactly with the source data. If not, either the source transformation logic or the control point logic is wrong and must be corrected.

FURTHER OPTIONS

Calculation Support to the Future System

The process as described above will by necessity have recreated all of the critical calculations (the Controlled Outcomes) for each of the source systems. These calculations will have been aggregated, normalised, optimised and validated for the Candidate Release for it to have been successfully validated and promoted.

These calculations will exist as IDIOM Decision Models, which can be natively executed via the Calculation Engine (either JAVA or Microsoft C#). There are several ways that the Calculation Engine can be used directly by a future System.

For the sake of clarity, the following suggestions do not require any further work on, or development of, the validated calculations.

Direct Call

If the platform allows, the Calculation Engine can be called directly in the address space of the calling function. If this approach is used, the calling system would be required to instantiate a DOM (or JDOM) that matches the Candidate Release PolicyXML and pass it into the Calculation Engine by reference.

Service Call – Caller Supplies Data

The service call would wrap the Calculation Engine in a service of the vendor's preference, which may be a queued service, a web service, or any other form of asynchronous or synchronous call to an external process.

The data would need to be provided by the caller in a format which matches the Candidate Release PolicyXML. The results are returned to the caller in the PolicyXML.

Service Call – Service Acquires the Data

The service call would wrap the Calculation Engine in a service of the vendor's preference, which may be a queued service, a web service, or any other form of asynchronous or synchronous call to an external process.

In this approach, the service would receive a policy key and a service request. The service would read the data from the System database and perform the calculation. The results could either be posted back to the database, and/or returned to the caller.

Further Benefits

Given any of the approaches above, it is plausible, even desirable, that the Accountable Person should directly own and manage the calculations on a go forward basis. IDIOM provides extensive support for SME driven unit and regression testing at scale, automated documentation, and audit and control of released artefacts, so that deployment of SME defined calculations directly into the system can be made both risk averse, and transparent and auditable, as prescribed by CPG235. This process is described by one IDIOM customer as 'nimble, continuous, perpetual'.

Audit and Remediation

Even given a perfect throw it is plausible that data will be found to be unfit for purpose in its new environment, or the new environment generates unexpected outcomes. In either case, regular (daily) 'business as usual' data validation and recalculation can provide peace of mind. This process can be implemented on behalf of any future system re-using the ISOL calculation assets already described.

The ISOL approach generates a register of precise item by item issues at Assurance Levels Two and Three that can be used to drive further analysis and remediation. It is a feature of the ISOL approach that the identification and analysis of Alerts as described in this document is a precursor step that is intended to progress seamlessly into analysis of remediation options, thence actions, and outcomes, both during the migration, and thereafter into production as required.

The ISOL approach fully supports remediation as follows.

When reviewed by SME's, the Alerts are recognised as situations that then require situation analysis as to cause and consequence, followed by analysis of remediation options. The options are subject to governance decision making by the Accountable Person to determine subsequent actions to achieve the remediation outcome.

The IDIOM Decision Manager Workbench™ is purpose built to support this data quality assurance cycle, which is summarised as follows:

- ✓ Identify the Situation;
- ✓ Situation Analysis (cause and consequence) of the Situation;
- ✓ Analyse Options;
- ✓ Governance Decision Point: Accountable Person to select and approve the preferred Option;
- ✓ Perform the Action as required by the Option;
- ✓ Confirm and document the Outcome.

ISOL asserts that this process will help the insurer to meet its obligations under CPG235. Further information on ISOL's approach to managing CPG235 can be found in the IDIOM whitepaper 'Data Integrity in Financial Services' available here.

IDIOM INSURANCE SYSTEM

We have referred to an ISOL supplied System. This System technology would be supplied in the form of an IDIOM Transaction Engine that is customised to the insurer's needs by directly ingesting the Bridge data and calculations.

We have described a process that allows for comprehensive product rationalisation. However, a reduced number of defined products does not of itself reduce the sum of the complexity that is required to support the portfolio.

This complexity may be reflected in data and data structures, supported features and functions, and calculation methods and process sequences. It can be stated with relative certainty that any future system has a known and finite ability to define and manage products. Any products that do not fit the available future system profile will require customisation of the future system if it is to support those products.

At the same time, the products themselves have differing demands for functional support, although all are equal in terms of their need for compliance. For instance, the on-sale products require more supporting functionality than the run-off products, which in turn require more than the archived products. The potential differences in required functional support can be used to reduce cost – why pay for 100% functional support for off-sale or archived products?

Furthermore, ISOL's anecdotal assessment is that the products that have reduced functional support requirements are also likely to have a greater calculation burden on a future System – that is, it may cost more to implement the least active products in the future System.

ISOL therefore proposes the IDIOM Insurance System concept. The implementation cost of each product in the IDIOM Insurance System will be marginal given the prior existence of a fully formed and validated Bridge.

What is the IDIOM Insurance System?

At the conclusion of Phase 2 as described in this document, the Bridge will contain complete product configurations; all required policy data as PolicyXML; and a Calculation Engine containing the complete set of Approved Formulas for the Controlled Outcomes.

These assets are the essential elements of a system; the remaining system functionality is relatively mechanical in nature and easily derived. The required functionality is known and finite, and includes:

- User authentication and authorisation
- Database infrastructure
- Console for administrative functions
- Forms for interacting with Policy data at the transaction level
- Batch jobs for applying scheduled changes to policy data, or for other routine changes and reporting
- Tightly managed integration components to external systems
- Generating customer and third-party documents and communications

- A management dashboard

Over the past 17 years IDIOM has developed all of these capabilities in generic form, which have been recently collated into a single execution framework that we have called the IDIOM Transaction Engine [ITE]. An AZURE PaaS version of the ITE was first used in production with a department of the Government of South Australia for state wide entitlement calculations in August 2016. In 2017, it was deployed on behalf of a major city council, and as a new direct to consumer insurance sales platform, both in NZ. A non-PaaS version is also available for use in VM or on-premise solutions.

The ITE is more fully described in the whitepaper 'IDIOM Transaction Engine – An Application to Manage Complex Business Entities' [available here].

It provides a complete infrastructure that is quite generic and without purpose when initially supplied. It has been purpose built to ingest verbatim the essential elements of the system as described by the Bridge, at which time it will perform the majority of the functions of a full insurance system. For the sake of clarity, these 'essential elements' of the system will be ingested without any further development effort on them.

IDIOM Insurance System Delivery

Assuming that ISOL proceeds to implement the Validated Product Database, then we typically provide some IDIOM Forms functionality to assist with viewing and managing various aspects of the Bridge.

This can very easily advance to an IDIOM Insurance System 'proof of concept'.

Actual delivery of an IDIOM Insurance System would need to be planned and costed when the requirements are known, in particular, exactly which products, and by implication what functionality, is to be managed by the IDIOM System. This could include a small set of residual products that cannot be easily migrated to a future system, through to larger sets of products, which for the sake of clarity, could extend to all of the insurer's products i.e. a full insurance system.

Assuming a fully functioning Bridge, then a running instance of the IDIOM Insurance System should be achievable in 100days elapsed from the point of requirements certainty.

IDIOM Insurance System SaaS

IDIOM would be pleased to discuss provision of the IDIOM Insurance System on a full-service basis (SaaS) if desired by the client. This would include all technical support and provisioning of run-time capacity.

Finis

John Salamito | CEO & Director | IDIOM Solutions

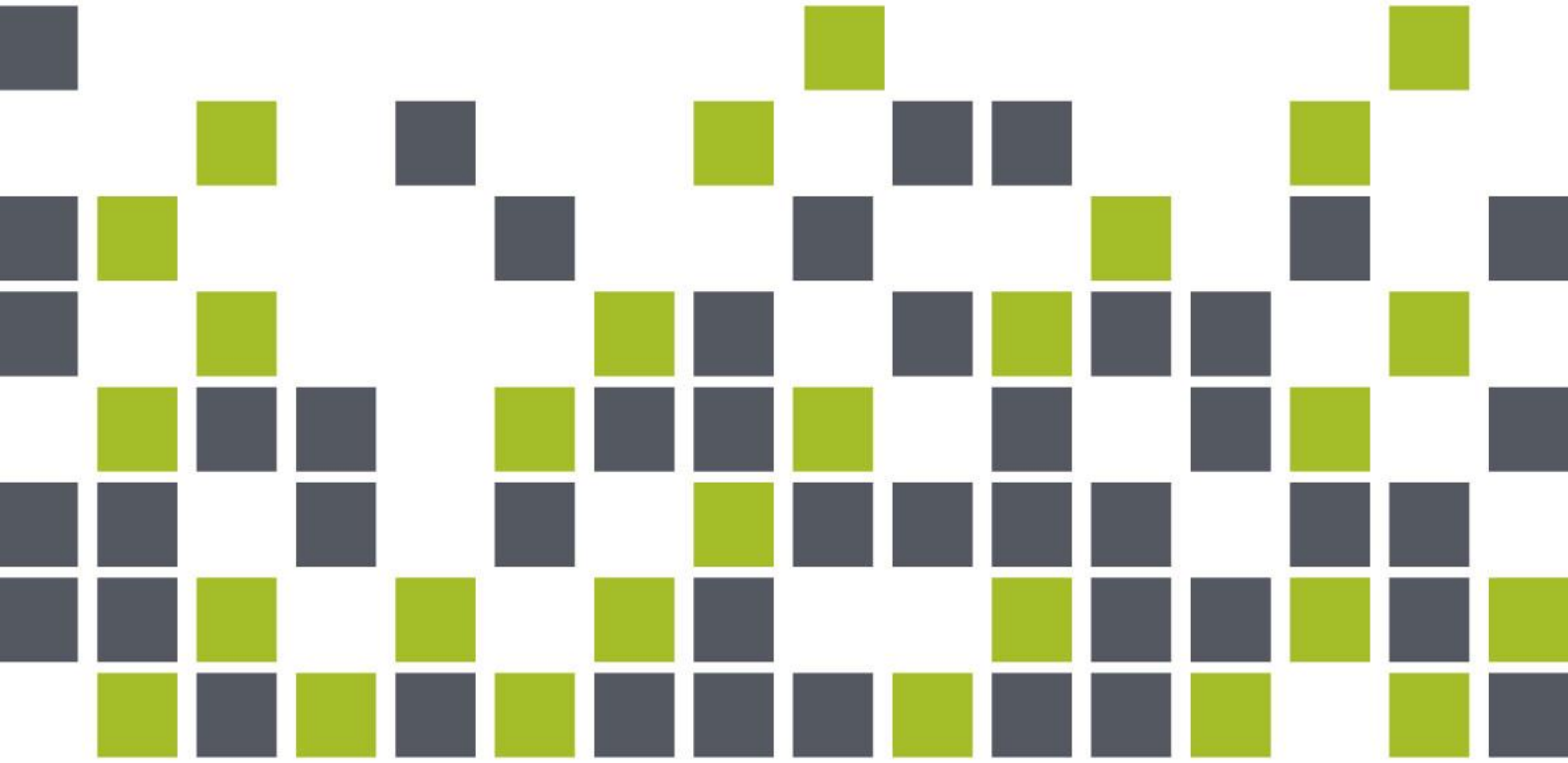
☎ +61 2 9968 4168 | +61 421 587 100 | ✉ Suite 16, Level Two, 357 Military Road, Mosman NSW 2088 Australia | @ john.salamito@idiom-solutions.com | idiomsoftware.com

Mark Norton | CEO & Founder | IDIOM Ltd

☎ +61 2 9968 4168 | +64 21 434 669 | ✉ Suite 16, Level Two, 357 Military Road, Mosman NSW 2088 Australia | @ mark.norton@idiomsoftware.com | idiomsoftware.com

Mark Midwinter | Director | IDIOM Solutions & IDIOM Ltd

☎ +61 2 9968 4168 | +61 419 211 192 | ✉ Suite 16, Level Two, 357 Military Road, Mosman NSW 2088 Australia | @ mark.midwinter@idiom-solutions.com | idiomsoftware.com



☎ +61 2 9968 4168 | idiomsoftware.com |
✉ Suite 16, Level 2, 357 Military Road, Mosman NSW 2088 Australia |