

KONDOR+ UPGRADES

2LA Solutions

Client Value Proposition

White Paper



Table of contents

1. OVERVIEW	3
2. KONDOR+ 3.0 NEW FEATURES	4
2.1. Functional Changes	4
2.2. Technical Changes.....	4
3. UPGRADE PROJECT	6
3.1. Purpose.....	6
3.2. Project Plan.....	7
4. PROJECT PHASES	8
4.1. New Features Analysis	8
4.2. Core Product Upgrade Analysis.....	8
4.3. Infrastructure Design, Validation & Implementation.....	8
4.4. Environment Migrations.....	10
4.5. Technical Core Product Regression Testing	10
4.6. Functional Core Product Regression Testing	10
4.7. Dress Rehearsals.....	11
4.8. Custom Impact Analysis	11
4.9. Custom Development Migration	16
4.10. Unit Testing.....	16
4.11. Integration Testing	16
4.12. Functional Testing	16
4.13. User Acceptance Testing	17
4.14. Go/No Go Meeting.....	17
4.15. Live Weekend	17
5. PROJECT ESTIMATES	18

1. *OVERVIEW*

Kondor+ as all software requires regular maintenance. That maintenance takes the form of minor upgrades for fixes and patches, and major upgrades bringing new functionality but also major changes such as data model changes, API, introduction of new technology.

Although this is unavoidable and in a sense is a protection of the client investment, such an upgrade can result in a painful project. As a company involved in such project before, and attuned to the need of the Kondor+ user community, we have produced this “lessons learned” document. Another investment was to develop specific tools to address the major findings of the lessons learnt.

In this document, we intend to present these lessons, and the help that we can provide you with to alleviate the underlying issue. According to the customer need, this help can be in terms of:

- Tools only
- Tools and services leveraging on these tools added value

These lessons may not be all applicable to you, but we hope that some will gain you valuable time and resources in your upgrade process.

2. **KONDOR+ 3.0 NEW FEATURES**

Before browsing the lessons learnt and the help that we can provide, let's see at a high level what has changed in Kondor+ 3.0.

The purpose of this section is to list the main changes brought by the Kondor+ 3.0 version which require a close attention by the customers. The goal is not to list every single change coming with 3.0.

The main changes are twofold, functional and technical.

2.1. **Functional Changes**

- Repo module redesign: this provides new functional features but also an impact on the database data model (former *RepoDeals* table split over *RepoDeals* and *RepoSecuSched* tables for example)
- Call Account module redesign
- MBS and CMO/ABS module redesign
- Loans & Deposits module enhancements
- Securities Lending module enhancements
- Revaluation module enhancements
- Securities Balance
- RisKatcher
- Deal versioning
- Global Forex Position
- Collateral management enhancements

2.2. **Technical Changes**

- *SUN Solaris 10* is required
- *Sybase 15.0* is supported
- *Hummingbird Exceed* support instead of *XVision* for remote display (UNIX to PC). *Citrix* is also supported.
- *kplus_dbo* Sybase user is the databases owner
- Full redesign of the database model:
 - o KplusGlobal
 - o KplusLocal
 - o Kustom
 - o KplusArchive
 - o KplusVersion
- *KreditNet* module embedded in Kondor+ until version 2.6 has been removed in version 3.0. Therefore the *KreditNet* database disappears
- Clients have to use views instead of table in order to interact with the database. As the data model has been completely re-designed this reduces the effort in terms of migration.
- Minimum hardware requirements are increased (see the Thomson Reuters recommendations)
- Report properties are in *XML* format (Financial report, etc.)

- New look & feel, user friendly native Microsoft Windows graphical user interface using the *QT* graphical library instead of the *SUN XView* library which is in care and maintenance.
- New procedure to deploy Kondor+: until Kondor+ 2.6 a Kondor+ installation or upgrade was performed via UNIX shell scripts. From Kondor+ 3.0 such this process relies on a JAVA GUI and XML allowing to completely set the system up.

3. UPGRADE PROJECT

3.1. Purpose

A Kondor+ migration project encompasses two critical phases:

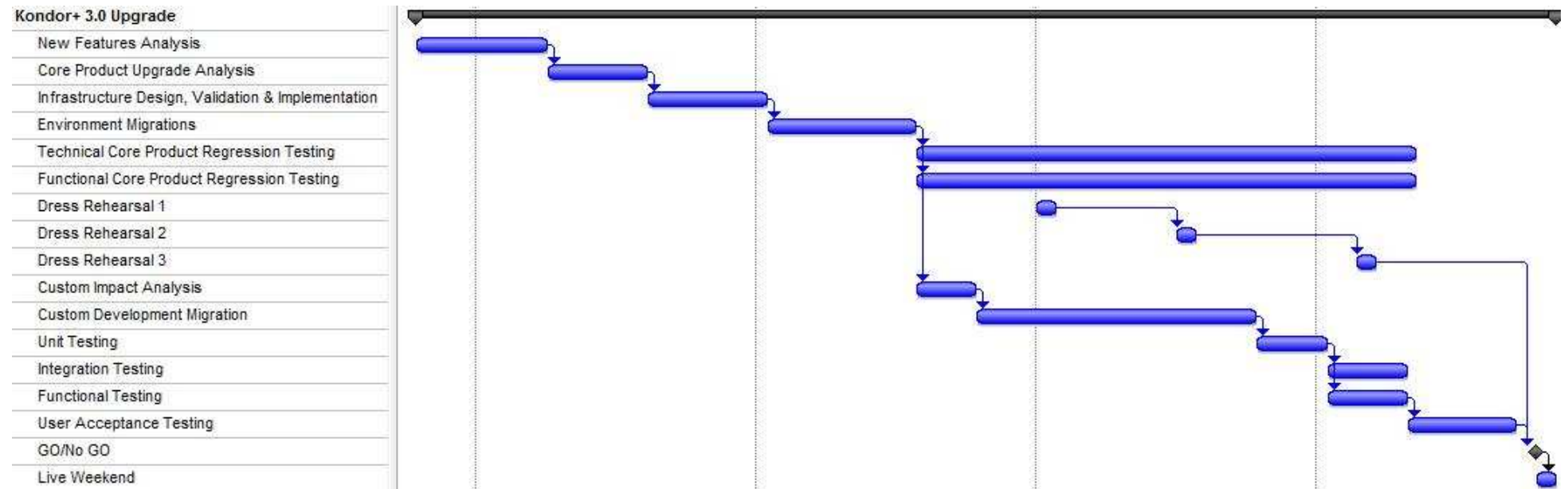
1. Budgeting
2. Planning and execution

This document will go through both aspects but starting with the planning and execution phase as the budgeting phase will be clearer in light of the project structure and its major parameters.

The project plan below is a generic plan to highlight the main phases of a Kondor+ 3.0 upgrade project.

3.2. Project Plan

The purpose of this plan is to show the main phases and their sequencing but not the durations as this is specific to each client. Therefore, do not take into consideration the tasks lengths.



All these phases will be detailed in the section below and then the project estimation in terms of time, cost and staffing will be presented.

4. PROJECT PHASES

For each project phase, a description is given and where applicable the extra help we could provide.

4.1. New Features Analysis

Phase Description

Thomson Reuters provides a “*What’s New?*” document with each Kondor+ release. Based on this high level document, clients can have a flavour of what the new features are in the Kondor+ 3.0 version.

It is important to assess which features will be used, as some choices may be made during the upgrade that will render the use of the feature more difficult. Once, some of them can be left to be implemented or activated until after the upgrade.

This analysis part can also be the opportunity to decide to decommission some customized developments which may be covered by the new Kondor+ version, such as some customized reports or custom windows which may no longer be useful.

4.2. Core Product Upgrade Analysis

Phase Description

An upgrade to a new Kondor+ version implies at least the following efforts:

- Hardware impact analysis
- 3rd party software impact analysis
- Core features to be tested (regression testing)
 - o Standard reports (Financial report, Cash Flow report, Gap Analysis, etc.)
 - o Deal Capture windows
 - o Pricing Windows
 - o Real time (quotations, yield and volatility curves, etc.)
 - o Revaluations

From that analysis, test cases can be written and executed during the *Functional and Technical Core Product Testing* phases presented below.

We strongly suggest organising functional workshops with Thomson Reuters on this subject, as not only does it mean getting the most value of the new version, it also helps understand the underlying changes and therefore being able to adjust more readily to the new version.

4.3. Infrastructure Design, Validation & Implementation

Phase Description

Kondor+ 3.0 is a major upgrade where:

- some key elements to the database model have been redesigned
- some new technology have been introduced
- some components have been re-architected

As a result, it requires additional resources, in line with the evolution of hardware standards. It is therefore important to specify and design the infrastructure that the new version will be running on. This is client specific as it is related to asset classes traded, functionality utilised, volumes, etc. and as such a “one-size-fit-all” infrastructure cannot be recommended. In this particular case, having developed the software or having already operated the software does not provide an accurate estimate of what the solution should be. So an actual evaluation based on the exact database, even of core functionality is essential. We therefore recommend running a core upgrade process in advance and obtaining key metrics comparison (Financials, etc.) between the two versions on the client upgraded database.

Once designed, the infrastructure should be validated as part of the technical core upgrade, running standard reports (i.e. Financial reports) and comparing the time to execute these with the current running time of these reports.

2LA Tools & Services

Having recognized the importance of this step, we have invested in developing a core automation tool, which needs to be customized to each client specific situation. This is an automated “single step” core upgrade process whatever the current Kondor+ source version (Kondor+ 1.9 or 2.x) that can be used numerous times, giving the project flexibility in terms of number of environments, but also in terms of migration strategy.

This automated upgrade tool taking care of the Kondor+ core upgrade is called *OneStep* and the main features are:

Fully automated upgrade

- One set of parameters before the upgrade launch
- Between any two versions
- No manual intervention
- Entry points for customization

Tailor made

- Source and target version
- Upgrade steps optimized for performance
- Scenario can rely on customer specificities

Flexible

- Scenario based
- Tasks execution in sequential or parallel modes
- Execution modes per task: auto, semi-auto, manual
- Possibility to resume the scenario execution from any point (task)

Centralized log management

- Global log
- Log per upgrade task

On that basis, one of our consultants could tailor made (onsite or offsite) your automated Kondor+ core upgrade. The result of this intervention is the availability of a scripted tool and a documented process.

The consultant would also be in charge of running the sample of standard reports in order to compare the performance before and after the upgrade process.

4.4. *Environment Migrations*

Phase Description

The project requires implementing and maintaining many upgraded environments. Each of these environments requires the development of a procedure to create it, to manage it (refreshing the data). This consumes resources and needs to be estimated.

Such an environment migration is a manual exercise with an increasing risk if the number of Kondor+ versions to be handled is more than one (for example migrating from Kondor+ 2.0 or Kondor+ 2.5 to Kondor+ 3.0). Moreover 3.0 additional loads and patches can contribute to this complexity.

Additional steps in the procedure to implement any 3rd party software necessary may be required, adding to the complexity.

It is key to envisage this step carefully, quite early, providing the project with the maximum flexibility and reducing the risk of errors being brought by environment differences, frequently the cause of wild goose chases.

2LA Tools & Services

Our “single step” automated core upgrade tool, called *OneStep*, would ease and secure the upgrade of all these environments. The client would only have to manage the deployment of its customizations.

4.5. *Technical Core Product Regression Testing*

Phase Description

The Core Product Upgrade Analysis phase presented above should provide documents presenting what test cases are for the technical regression testing.

This should encompass for instance running the standard reports prior to upgrade, running the test cases and getting their performance, running the upgrade, running the test cases on the upgraded database and validating the performance.

Such a performance validation should also be performed for the Kondor+ handlers, servers and night batch.

4.6. *Functional Core Product Regression Testing*

Phase Description

The Core Product Upgrade Analysis phase presented above should provide documents presenting what test cases are for the functional regression testing.

The report comparison here is key again. Whereas the above phase was focused on the performance validation, here, comparing reports prior to upgrade and post upgrade is to validate the financial numbers (Financial report, Gap Analysis report, etc.).

This is the minimal validation but other validation aspects have to be considered like: deal entry, pricing tools, real-time position keeping or night batch.

4.7. *Dress Rehearsals*

Phase Description

This usually requires several attempts to validate the Live Weekend procedure.

The upgrade run book should be under the control of the project manager to check that the whole upgrade scenario is successfully performed and that at the end the system is up and running as expected.

This is also the opportunity to define a rollback scenario and validate it.

2LA Tools & Services

Our *OneStep* tool can be of real benefit here to reduce the operational risk of the upgrade itself, but also the complexity of the Dress Rehearsals.

4.8. *Custom Impact Analysis*

Phase Description

This phase is one of the most critical phases in a Kondor+ migration for a client as it should at least cover the following aspects:

- Analyze the impacts on the in-house developments regarding:
 - o Custom Windows
 - o Customized reports: OpenReports and KSQL reports
 - o Interfaces:
 - SQL stored procedures
 - Import / Export files
 - o Scripts: SQL and UNIX
 - o Other: specific tools may have been developed in C/C++/Java/etc and interacting with the Kondor+ database.
- Based on that analysis, test cases have to be written in order to test all the changes to be done during the Custom Development Migration
- Customized developments clean-up: some historical developments may not be used any longer and it could be the opportunity to identify them and get rid of them. This would prevent from migrating what is unused, meaning reducing the maintenance cost (migration) and could reduce the *Custom Development Migration* phase.

The mistake here is to believe that understanding of the code is required in order to make the necessary change. Experience has shown that this is not the case for 90% of all developments, where a change *strategy*, i.e. a “by default” way of correcting the issue can be defined and successfully applied.

Defining such a change strategy, although can be costly in the early part of the project, delivers tremendous benefits during the execution phase in reducing the risk of interpretation, misunderstanding, etc. This migration strategy should allow either for a vertical approach, dealing with a specific issue across all code elements, or horizontal, allowing to deal with all upgrade impacts in a specific code element.

The difficulty is to carry a deep exhaustive analysis that allows to document the issues and define a strategy for each type of impact, across all the custom code elements.

2LA Tools & Services

Recognising the importance of this step, we have developed a tool that will analyze the customer code and provide a detailed view of the upgrade impact on it, file by file, change by change (Kustom Impact Analyzer tool).

In addition for the 10% code that will have to be understood, we deliver, bundled with the KIA, a customer centric documentation that will facilitate the analysis (for the specific client start and end upgrade versions).

On this basis, we can deliver a *migration strategy* report that identifies all the changes and recommends an approach. One of our consultants would build such a report from the KIA output and propose a tailor made approach to migrate your customized developments. This approach would be as indicated above, vertical, horizontal or a mix of both according to your specificities. This report delivered could also present a strategy per asset classes (bond specialist for instance), per technical type of elements (open reports), etc.

Regarding the KIA product, you can find below a few snapshots including the embedded tailor made K+ documentation:

- This first one shows all the procedures and per procedure what the impacts are. Moreover, these impacts are displayed within the source code.

The screenshot displays the 'Kustom Impact Analysis' interface for the 'K3S_RepoDeals_CF' module. The interface includes a navigation menu on the left with various modules, a central table of changes, and a SQL source code view at the bottom.

Reference	Line	Action	Change
RepoDeals.Basis	line 43	UPC	Choice [Basis]: values have changed[Table Doc] [Choice Doc]
RepoDeals.Bonds_Id	line 33	MOV	Moved: RepoDeals.Bonds_Id --> RepoSecuSched.Bonds_Id [Table Doc]
RepoDeals.DealType	line 29	UPC	Choice name has been changed: BuySell --> RepoDealType [Table Doc] [Choice Doc]
RepoDeals.Equities_Id	line 34	MOV	Moved: RepoDeals.Equities_Id --> RepoSecuSched.Equities_Id [Table Doc]
RepoDeals.Indexation	line 44	UPC	Choice [FixedFloating]: values have changed[Table Doc] [Choice Doc]
RepoDeals.Papers_Id	line 35	MOV	Moved: RepoDeals.Papers_Id --> RepoSecuSched.Papers_Id [Table Doc]
RepoDeals.WeightedAmount	line 37 line 65	MOV UPN	Moved: RepoDeals.WeightedAmount --> RepoSecuSched.WeightedAmount This field now accept Null value. [Table Doc]
RepoSchedule.RepoDeals_Id	line 60	UPN	This field now accept Null value. [Table Doc]

DEL: Deleted field - **MOV:** Moved - **REN:** Renamed
NEW: New field - **UPD:** Updated datatype - **UPC:** Updated Choice - **UPN:** Updated AllowNull flag - **REV:** Section to review

```

24      SELECT
25          Source = @Strategy,
26          Status = D.DealStatus,
27          Deals_Id = D.RepoDeals_Id,
28          sourceInsID = D.RepoDeals_Id,
29          DealType = D.DealType,
30          Folders_Id = D.Folders_Id,
31          Cpty_Id = D.Cpty_Id,
32          Cpty_Id Issuer = null,
33          Bonds_Id = D.Bonds_Id,
34          Equities_Id = D.Equities_Id,
35          Papers_Id = D.Papers_Id,
36          Currencies_Id_1 = D.Currencies_Id,
37          WeightedAmount = D.WeightedAmount,
38          Countries_Id_1 = C.Countries_Id,
39          PaymentDate = S.PaymentDate,
40          CashFlow1 = S.CashFlow,
41          MaturityDate = D.MaturityDate,
42          MaturityRate = D.MaturityRate,
43          Basis = D.Basis,
44          Indexation = D.Indexation,
  
```

This can be done module per module but also from a functional point of view, i.e. table per table or choice per choice. Of course only the elements having an impact are listed.

- This second snapshot below shows the detailed documentation (full database data mapping field by field) which allows to browse through all the Kondor+ tables, fields and choices. A Google like Search Engine allows to look for any object within the database. You can jump to the documentation Search Engine from the KIA presented above.

The screenshot shows a search for 'repo deal sched' in the Kondor+ Database Documentation. The search results list 10 items, including RepoSchedule, RepoSecuSched, and various templates. The 'RepoSecuSched' entry is highlighted, and its detailed documentation is shown in a separate window.

RepoSecuSched

Heading	Money Market
Display name	Repo Securities Schedule
Number of Fields	39
Comments	Allows you to capture a repo Deal s set of securities

Legend: HOC: No Changes; DEL: Deleted field - MOV: Moved - REN: Renamed; NEW: New field - UPD: Updated datatops - UPC: Updated Choice - UPN: Updated AllowNull flag

Impact	Table name	Field name	Datatype	Null	Table name	Field name	Datatype	Null	Comments
NEW	RepoDeals	Accrued	float	Yes	RepoSecuSched	Accrued	float	Yes	Kondor+ automatically displays the accrued interest of the bond when the bond name is specified. Accrued interest is calculated on the value date. The value date is calculated by the nightly batch process.
NEW UPN	RepoDeals	Amount	float	No	RepoSecuSched	FaceAmount	float	Yes	Specifies the nominal value of securities exchanged. This field is activated for bond underlyings. This is calculated automatically when the Quantity field is specified using the following formula: (Quantity x Face Value of the bond)
NEW					RepoSecuSched	BondsSamples_Id	int	Yes	Id of the bonds sample
NEW	RepoDeals	Bonds_Id	int	Yes	RepoSecuSched	Bonds_Id	int	Yes	Id of the bond
NEW HOC UPN	RepoDeals	CapturedMarketValue	choice	No	RepoSecuSched	CapturedMarketValue	choice	Yes	Indicates whether the market value was captured by a price or a yield.
NEW UPN	RepoDeals	ClearingNodes_Id	int	No	RepoSecuSched	ClearingNodes_Id	int	Yes	Allows you to specify the clearing mode used for the deal. This field is mandatory if IsUsed = YES.
NEW	RepoDeals	ConversionRate	float	No	RepoSecuSched	ConversionRate	float	No	If the deal currency is the same as the security currency, the conversion rate (exchange rate) is equal to 1. If the currency is different from the currency assigned to the bond on equity, K+ automatically enters this rate.
NEW	RepoDeals	CurrentFactor	float	Yes	RepoSecuSched	CurrentFactor	float	Yes	Specifies the current factor of the MBS/ABS (ex Pool Factor) This is the ratio of the outstanding principal balance (as of the trade date) to the original principal balance (as of the last record date before the trade date).
NEW					RepoSecuSched	DividendPart	float	Yes	Dividend Part is used to compute the lines of fee for the repo on equity.
NEW					RepoSecuSched	EndDate	datetime	Yes	Specifies the end date of the chosen cash flow.
NEW					RepoSecuSched	EquitiesSamples_Id	int	Yes	Id of the equities sample.
NEW	RepoDeals	Equities_Id	int	Yes	RepoSecuSched	Equities_Id	int	Yes	Id of the equity.
NEW					RepoSecuSched	ForwardConversionRate	float	No	The forward conversion rate.
NEW	RepoDeals	ForwardPrice	float	Yes	RepoSecuSched	ForwardPrice	float	Yes	Specifies the price of the bond used in the collateral of a Repo on the maturity date. The price is automatically calculated when a fixed rate or a fee is specified as the payment type.
NEW	RepoDeals	ForwardYield	float	Yes	RepoSecuSched	ForwardYield	float	Yes	Displays the Forward yield of the bond. The yield is automatically calculated when a fixed rate or a fee is specified as the payment type.

- This third snapshot presents the Import/Export files analysis based on the *KplusFeed* or *KML* (Kondor+ XML) formats supported by KIS and TradeKast. This feature is embedded within the KIA tool.

Impact	Table name	Field name	Table name	Field name	Field comment	Impact comment
UPC	ReposDeals	Basis	ReposDeals	Basis	Specifies a basis for the calculation of interest. By default, the money market basis is automatically displayed as defined in the Calculation Parameters window for the currency of the deal. This can be changed.	
UPC	ReposDeals	CompoundAverageFrequency	ReposDeals	CompoundAverageFrequency	If indexation of deal is Floating rate then you can specify Compound interest calculated at Annual, Semi-Annual, Quarterly, Monthly, or Daily compounding frequencies. However, Compound Frequency field is deactivated for fixed repo deals (added 1.7).	
UPC	ReposDeals	DealType	ReposDeals	DealType	Specifies whether the deal is a buy or sell deal.	
UPD, UPN	ReposDeals	DownloadKey	ReposDeals	DownloadKey	An alphanumeric string used to download or transfer information between Kondor+ and another system.	
UPC	ReposDeals	Indexation	ReposDeals	Indexation	Specifies whether the repo deal is based on a fixed or a floating rate. This field is activated when rate is specified in the Payment Type field.	
NEW, UPD, UPN	ReposDeals	CapturedMarketValue	ReposSecuSched	CapturedMarketValue	Indicates whether the market value was captured by a price or a yield.	
MOV	ReposDeals	ForwardPrice	ReposSecuSched	ForwardPrice	Specifies the price of the bond used in the collateral of a Repo on the maturity date. The price is automatically calculated when a fixed rate or a fee is specified as the payment type.	
MOV, UPN	ReposDeals	GrossAmount	ReposSecuSched	GrossAmount	The gross amount of the deal is the market value of the securities exchanged at value date. This is automatically calculated using the following formula: For a bond [(Price + Accrued) x Face Amount / 100] For an equity [Price x Quantity]	
MOV, UPN	ReposDeals	Haircut	ReposSecuSched	Haircut	Specifies the haircut of the deal. By default, the haircut is 100 percent, unless the hedge ratio of the bond or equity has been specified in the Data Manager. (changed 1.7).	
NEW, UPN	ReposDeals	IgnoreCouponPayments	ReposSecuSched	IgnoreCouponPayments	Specifies whether coupon payments are used in the calculation of the forward amount of the Repo collateral, and whether the cash flow is integrated in the cash flows of the deal. This field is activated for bond deals.	
NEW, UPN	ReposDeals	Price	ReposSecuSched	Price	Specifies the price of the repo.	
NEW, UPN	ReposDeals	Quantity	ReposSecuSched	Quantity	Specifies the number of the repo contracts.	
MOV, UPN	ReposDeals	WeightedAmount	ReposSecuSched	WeightedAmount	The weighted amount is the cash amount of the deal and is used to calculate the interest. The weighted amount is calculated as follows: [Weighted Amount = Gross Amount/Haircut x 100]	
NEW FIELDS						
NEW			IsCLS		Is CLS (Continuous Linked Settlements) mode or not.	
NEW			Equities	Equities_Id_DataSplit	Used for equitySplit operation. Id of the old instrument before the split. 0 if not relevant.	
NEW			Equities	ISINCode	ISIN Code of the Equity	
NEW			Folders	Entities_Id	Entity owning the Folder	
NEW			ReposDeals	AdjustMaturityDate	Indicates if the Maturity Date must be adjusted in the schedule.	
NEW			ReposDeals	Adjusted	Indicates if the dates used to calculate interest payments are the same as the dates when the interest is paid. Activated: roll convention determines the calculation AND payment dates of interest. Deactivated: determines the interest payment date	
NEW			ReposDeals	BrkFrdInterpolation	Specifies whether the floating rates are interpolated on broken coupon periods for FRNs.	
NEW			ReposDeals	Correlation_Quanto	FX/IR correlation used for Quanto calculation.	
NEW			ReposDeals	Days	Day of the month to set the Payment Date in the Other Roll Convention.	
NEW			ReposDeals	DeliveryMode	Specifies the delivery mode of the repo deal.	
NEW			ReposDeals	DepositBatchDate	Field used for the computation of the Deposit Position in the Batch.	
NEW			ReposDeals	EndDateRollConvention	Roll Convention of the End Date (used only if Roll End Date is	

The value created is both in the deliverables and in the approach: the approach structures the project and makes it more likely to succeed. The deliverable in the form of the migration strategy will reduce the migration effort to a third of what it would have been otherwise, while at the same time making it exhaustive. For example, on a large implementation, going through all reports and amending them could have taken around 12 man/month. This would be reduced to less than 4 months using our migration strategy deliverable. We estimate that our services will allow customers to reduce their custom code migration by 3 times in average.

The documentation will not only provide value during the upgrade, but will also be useful after the upgrade for any new customized development or related analysis.

4.9. *Custom Development Migration*

Phase Description

This phase encompasses the execution of what has been analyzed in terms of impacts on the customized developments.

2LA Tools & Services

On the basis of the strategy report, we are happy to be involved in the execution at many levels:

- Provide the fully browsable analysis tool that will help developers focus on the code requiring changes
- Provide the full migration strategy that will describe what change needs to be applied intelligently for that type of issue
- Participate in the execution of these changes
- Fully undertake the execution of these changes, delivering a 3.0 compliant code

4.10. *Unit Testing*

Phase Description

This is performed by the developers performing the Custom Code Migration. Test cases should be written in order to know what has been tested but also to ease the repetition of these tests.

2LA Tools & Services

If we were part of the Custom Code Migration phase, unit test cases would be written to highlight the changes and define what would have to be tested.

The unit test cases would be based on details provided by the KIA and the tailor made documentation tools.

The corresponding tests would be conducted and documented.

4.11. *Integration Testing*

The testing team has to perform these tests based on the test cases written during the Custom Impact Analysis phase in order to check that the changes do not have any technical side effects and that the application is running.

4.12. *Functional Testing*

Phase Description

The testing team has to perform these tests based on the test cases written during the Custom Impact Analysis phase in order to check the results that the results are correct and that the changes do not have any functional side effects.

4.13. *User Acceptance Testing*

Phase Description

This phase is the functional acceptance of the new Kondor+ version by the end users.

The end users will test what they use on their daily activity for the core and customized features/tools.

This phase is critical as it should provide the acceptance of the system by the end users and consequently guarantee that the system behaves as expected. But by experience, it is always difficult to have end users really involved and motivated to perform the proper tests. This means that they should be driven by test cases listing what they should test and provide feedback on each of these tests. This is why User Acceptance test cases have to be built all along the project during the previous phases.

Unless this is already documented within the bank, this means that user interviews have to be performed to get an exhaustive list of what they use within Kondor+ in terms of core and customized modules.

4.14. *Go/No Go Meeting*

This meeting takes place to decide if the sponsors and the stakeholders within the bank agree to be live in light of feedback from the end users and the Dress Rehearsals.

4.15. *Live Weekend*

This is the real migration part on the production system(s).

5. PROJECT ESTIMATES

Having in mind the previous pages, one of the trickiest aspects is to evaluate the project cost as it relies on many aspects, like:

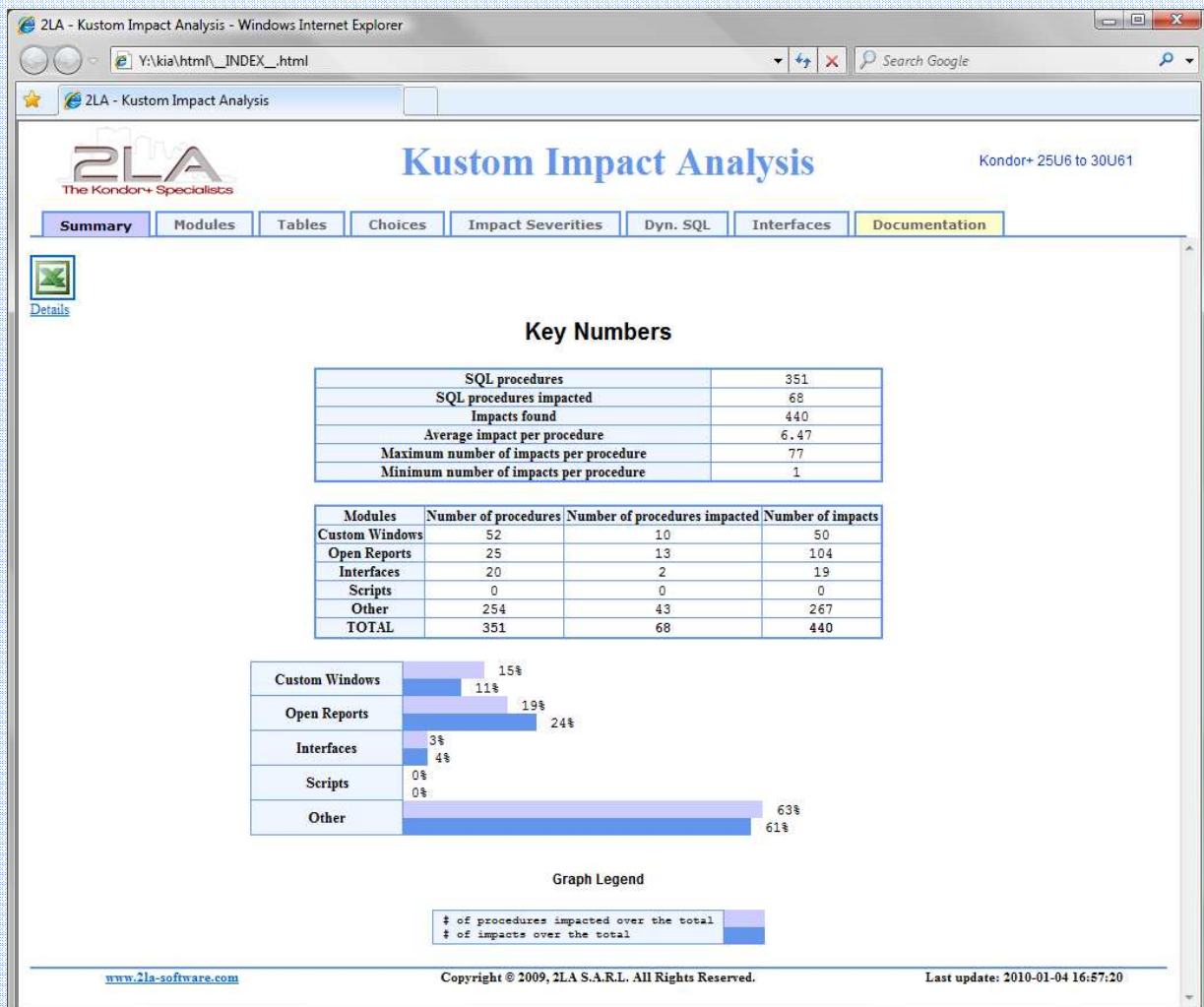
- What has changed in the new version
- What these changes imply in terms of impacts on the client customized developments
- What are the phases to be conducted within the project
- How many resources will be necessary to analyze and perform the migration (custom code, environments, etc.)?
- How to staff the testing team
- Etc.

2LA Tools & Services

On the many elements seen before, we can provide additional help in sizing the most variable part of the upgrade: the custom code migration.

Based on our KIA tool, one of our consultants can perform an exhaustive pre-analysis on the customized code migration which will provide:

- Clear metrics of the upgrade impact on the custom code base
- A proposal in terms of planning
- A proposal on how this migration phase should be staffed



We also believe that the other support we can provide do make the budgeting process easier:

- “single step” automated core upgrade tool called *OneStep*
- Custom development migration strategy
- Custom development migration execution