



# Risk Management in Offshore IT Projects

ZEBRA

LE COMPTE COURANT QUI VA PLUS LOIN

Siroco.NET



# Index

1. Introduction
2. Project Example Context
3. The main Phases in an Offshore IT Project
4. Risks in the Approach Phase
5. Risks in the Offshore Partner Selection Phase
6. Risks in the Execution Phase



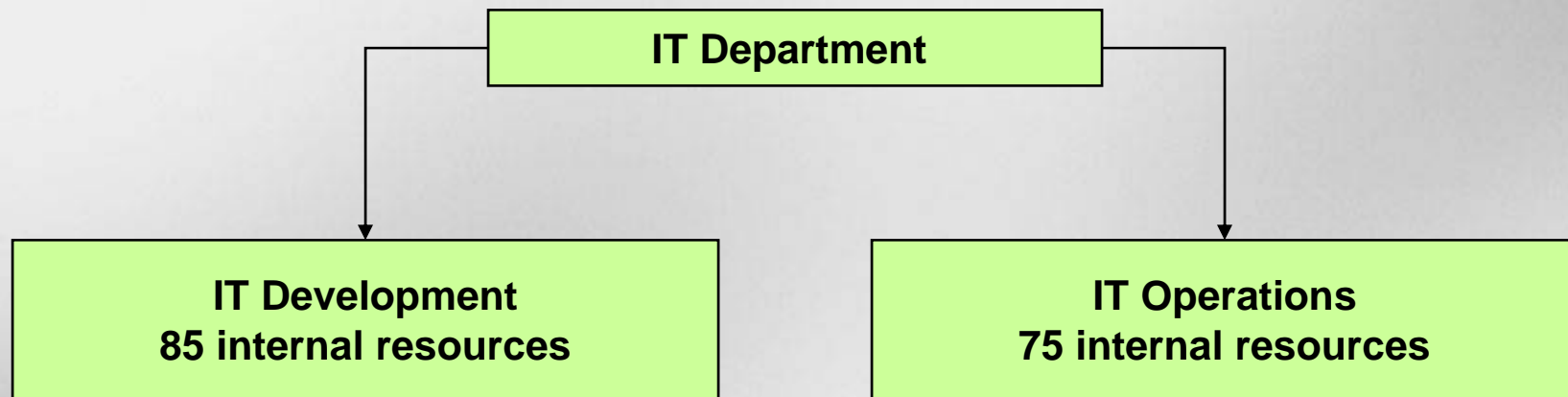
# 1- Introduction

- The risks of large IT projects are well known and there are several mitigation strategies to cope with them and succeed executing the project on time, on budget and on scope.
- For large IT offshore projects, there is the added complexity of the offshore model and its implications: remote teams, new infrastructures, different timezones, different cultures...
- The purpose of this presentation is to present the main risks associated with the management of a large IT offshore project, sharing our experiences and lessons learned at the Siroco project at the Banque et Caisse d'Épargne de l'État.
- The document is structured in 5 chapters:
  - An introduction of the project context
  - The phases of an Offshore project
  - The risks in the Approach Phase
  - The risks in the Provider Selection Phase
  - The risks in the Execution Phase



## 2- Project Example Context – the BCEE

### BCEE IT



- The IT internal resources represent **10%** of the total workforce
- In addition, there are around **30 and 50** external resources



## 2 - Project Example Context – the BCEE

### *Opportunity-based outsourcing*

- For every IT project, outsourcing is an option to be considered between others
- Examples of Outsourcing Projects at BCEE
  - Externalisation of the Installation and maintenance of PC's
  - Development and maintenance of our Front-End Internet Solution
  - Creation and hosting of our web site and the e-mail externes addresses.
  - Monitoring of our Branch Network lines.
  - Externalisation of the Maintenance of our Leasing Application
  - **Offshoring of the Siroco Migration Project to HP India**



## 2 - Project Example Context – Siroco



- Siroco is an integrated retail application to support client data, accounts, payments, transactions and cards.
- The development started in 1995 and the first rollout was in 1997. Since then, Siroco has constantly evolved to meet the new market requirements.
- Functional wise, Siroco perfectly meets the user requirements and regularly receives the best scores from the user ratings.
- Technical wise, Siroco was developed as a service oriented application with 250 rich windows developed with C and Foundation (case tool) and 800 cobol services (400.000 lines of code).
- The Foundation case-tool represented an important risk as it was a proprietary tool at the end of its lifecycle.





## 2 - Project Example Context – The Migration

### The Migration Principles

1. Iso functional Migration with the same look and feel.



## 2 - Project Example Context – The Migration

### The Migration Principles

1. Iso functional Migration with the same look and feel.
2. Only the front-end is rewritten. The existing Siroco services in the mainframes should be reused and the data model remains unchanged.



## 2 - Project Example Context – The Migration

### The Migration Principles

1. Iso functional Migration with the same look and feel.
2. Only the front-end is rewritten. The existing Siroco services in the mainframes should be reused and the data model remains unchanged.
3. Phased Migration to avoid a big bang testing and rollout



## 2 - Project Example Context – The Migration

### The Migration Principles

3. Phased Migration to avoid a big bang testing and rollout

**Block 1- Client Data (54 windows)**

**Block 2 – Accounts, Products, Cards (106 windows)**

**Block 3 – Transactions (62 windows)**

**Block 4 – Complex Transactions ( 28 windows)  
migrated by the BCEE**



## 2 - Project Example Context – The Migration

### The Migration Principles

1. Iso functional Migration with the same look and feel.
2. Only the front-end is rewritten. The existing Siroco services in the mainframes should be reused and the data model remains unchanged.
3. Phased Migration to avoid a big bang testing and rollout
4. Coexistence between the old and the new versions in Production to facilitate the rollout strategy



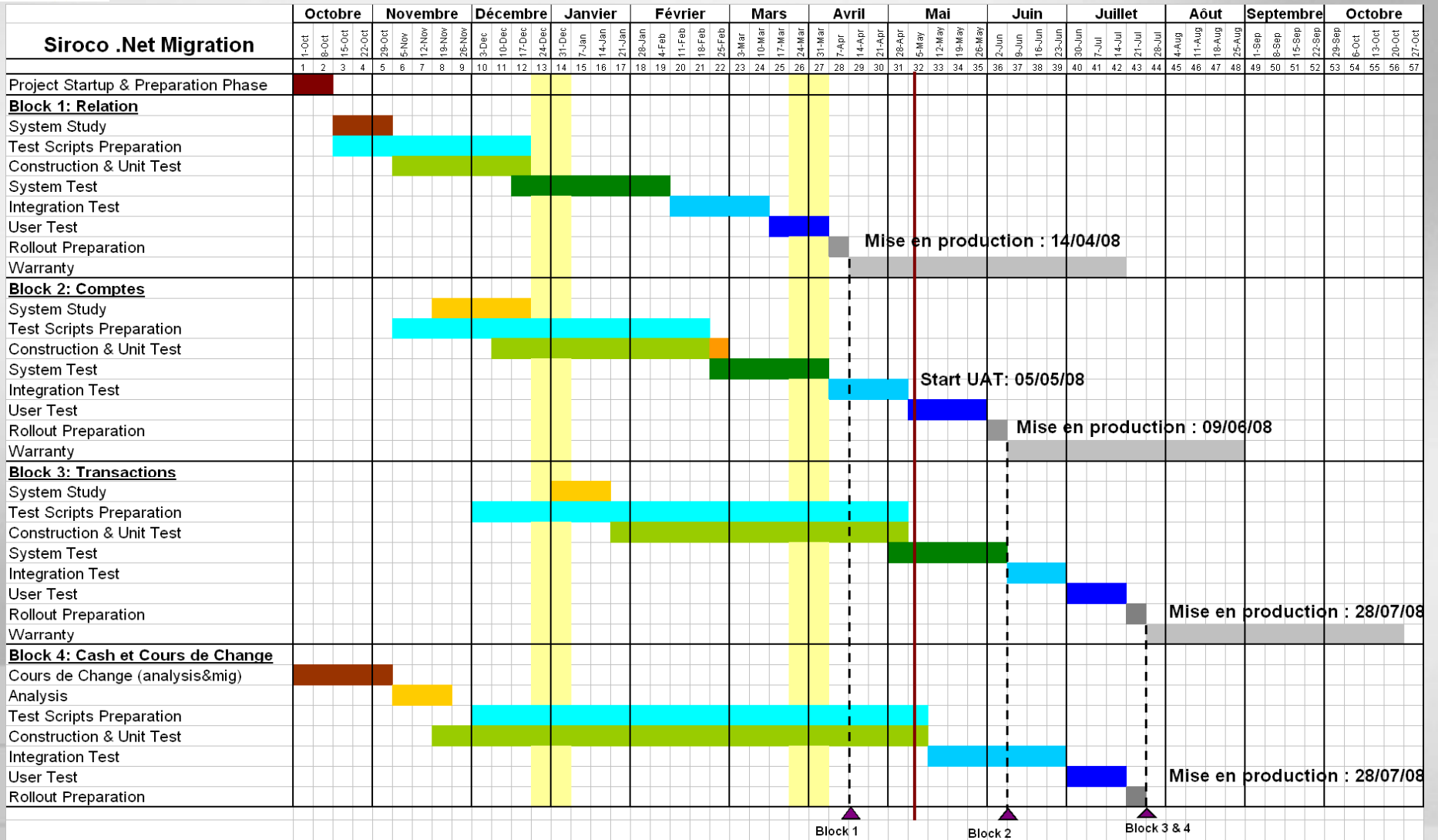
## 2 - Project Example Context – the Facts

- The BCEE outsourced the migration of the Siroco front-end to **HP India**
- The duration of the project was fixed to **10 months** to minimize the impact of the functional freeze to the users.
- The total size of the project was **10.000 mandays**:
  - ✓ HP: 6.000 mandays
  - ✓ BCEE IT: 3.200 mandays (2.000 for integration testing)
  - ✓ BCEE Users : 850 mandays
- The contract was negotiated with a fixed prize, a fixed duration, several SLA's, a guarantee period and several penalties for:
  - ✓ non respect of the delivery dates
  - ✓ non respect of coding standards
  - ✓ non respect of the SLA's for the production support





## 2 - Project Example Context – the Results

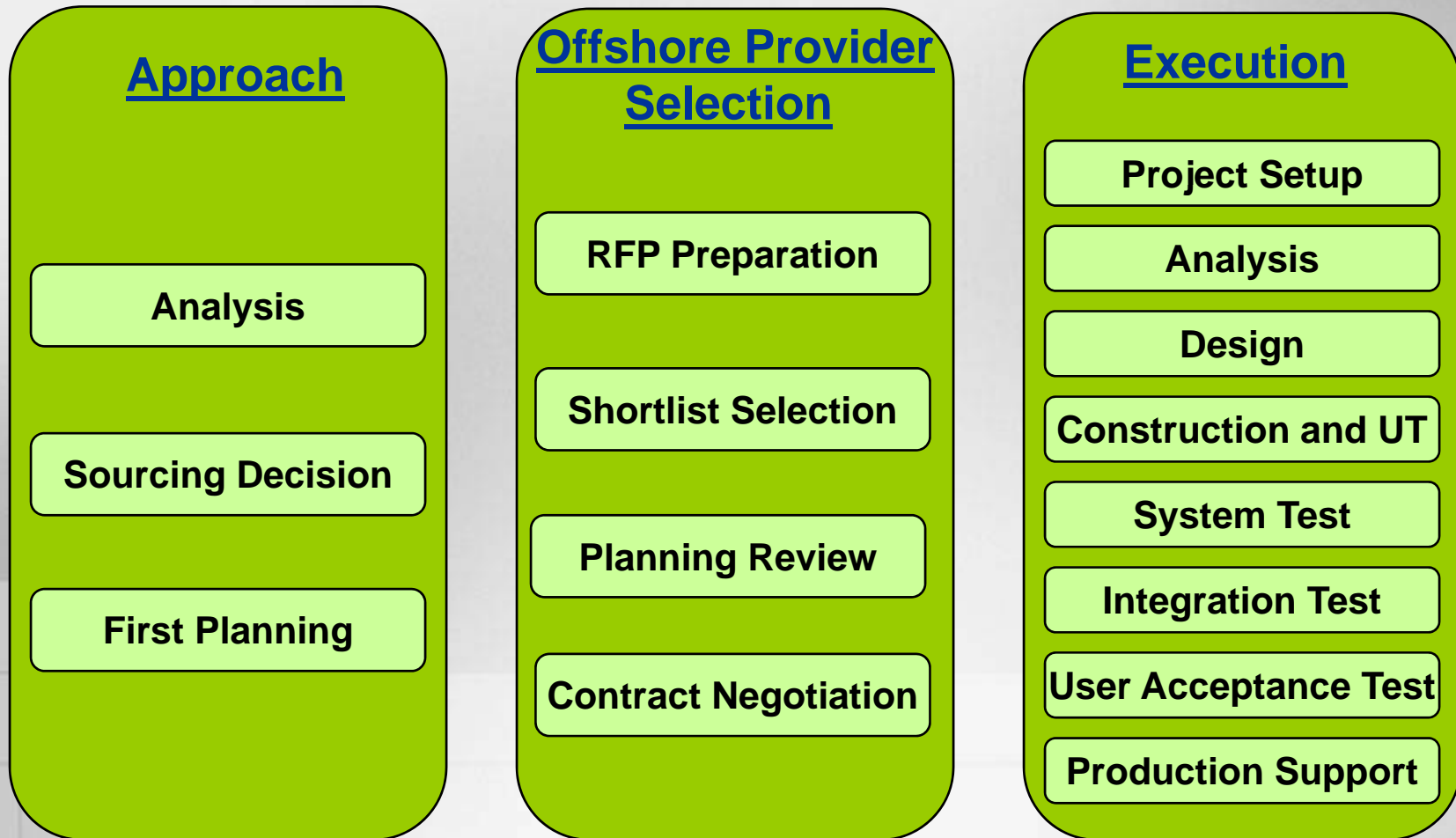


After 10 months of hard work, the last block was successfully rolled out in Production



## 3 – The Phases in an Offshore Project

The main risks in every IT project are not to finish the project on time, on budget and/or on scope. These high level risks can materialise if other risks are not properly managed in each of the phases:





## 4 – Risks in the Approach Phase

- **Choose the wrong sourcing for the project (offshore/onshore)**
  - ✓ Offshoring is not always the best choice, it depends on many factors (cost, availability of local resources, know-how, previous experience, etc).
  - ✓ If it is the first time that you offshore, then choose a project where the scope is very well defined (both functional and technical).
  - ✓ In the Siroco project, offshoring was a good choice because of the nature of the project (technical migration), the lack of .net resources in Luxembourg and the cost factor.

### Approach

**Analysis**

**Sourcing Decision**

**First Planning**



## 5 – Risks in the Offshore Provider Selection

### ➤ Scope is not well defined

- ✓ Be as much detailed as possible in the RFP document, with all your expectations in terms of planning, SLA's, quality...

### ➤ How to choose the best offshore provider?

- ✓ Preselect only 4 or 5 providers to send the RFP.
- ✓ In order to gain time and enforce concurrency make a common information sessions « information week » where all providers can openly put their questions in front of their concurrents.
- ✓ Shortlist just 2 (max 3) and open negotiations with them.
- ✓ Verify the experience of the local and offshore teams.
- ✓ Ask precise questions about the approach for specific important tasks (e.g. how you do the knowledge transfer?)

### Offshore Provider Selection

RFP Preparation

Shortlist Selection

Planning Review

Contract Negotiation



## 5 – Risks in the Offshore Provider Selection

- **The planning is not realistic**
  - ✓ Do not propose a plan that you are not able to keep.
  - ✓ Openly ask if they think they can stick to the proposed plan.
- **The contract is not balanced**
  - ✓ Ask for a contract type example during the selection process to prepare the negotiation phase.
  - ✓ A good contract has to be balanced for both sides, and responsibilities have to be clear and fair.
  - ✓ Again, detail as much as possible the planning and the SLA's.
  - ✓ In case of a fixed contract you have to define penalties.
  - ✓ Foreseen all possible scenarios and limit the possibilities of the exit-clauses.





## 6 – Risks in the Execution

### ➤ Technical infrastructure is not available

- ✓ Start as soon as possible the setup of the technical. infrastructure and do not underestimate the security issues.
- ✓ Define checklists and escalation procedures to apply when the communication are down.
- ✓ Provide high availability to the infrastructure for the offshore teams.
- ✓ Beware with the « anchors in the egyptian sea » during the duration of the project.



### ➤ Communication and cultural issues

- ✓ Meet the project leader of the offshore team early in the project.
- ✓ It is preferably to have a local coordinator from the offshore team.
- ✓ The local coordinator should join asap, but beware that the visa procedures are always a nightmare.
- ✓ Bring offshore teams for the knowledge transfers (and ask them to write the documentation that you will later review).
- ✓ In our case, daily conference calls with the offshore teams much helped to speed up the resolution of open issues.

### Execution

Project Setup

Analysis

Design

Construction and UT

System Test

Integration Test

User Acceptance Test

Production Support



## 6 – Risks in the Execution

### ➤ Quality issues

- ✓ Define penalties based on quality checklists.
- ✓ Provide to the offshore team quality standards and code examples.
- ✓ Plan an early deliverable for quality checkup and before the offshore factory is at full steam.
- ✓ Plan regular quality reviews.
- ✓ Define quality as one of the acceptance criterias.

### ➤ Resources availability (onshore and offshore)

- ✓ Be sure your provider is able to scale up resources
- ✓ Analogously, for your own resources, plan in advance to have margins to scale up.
- ✓ Quickly identify and correct resources bottlenecks.
- ✓ Involve Users in the early stages of the project (e.g. for knowledge transfer and Integration Test).

### Execution

Project Setup

Analysis

Design

Construction and UT

System Test

Integration Test

User Acceptance Test

Production Support



## 6 – Risks in the Execution

### ➤ Testing issues

- ✓ Test, test and test.
- ✓ Prepare and execute test scenarios in each phase:

Phase	Responsible	Support
Unit Test	Offshore Programmer	Unit Test Conditions
System Test	Offshore Test team	Offshore Test Cycles
Integration Test	IT	Integration Test Cycles
UAT	Users	User Test Scenarios

- ✓ Ideally provide 3 different test environments for System Test (offshore), Integration Test and User Acceptance Test.
- ✓ Define and monitor specific KPI's for testing (# incidents in System Test, Integration Test and UAT)

### Execution

Project Setup

Analysis

Design

Construction and UT

System Test

Integration Test

User Acceptance Test

Production Support



## 6 – Risks in the Execution

Example of the Risk follow-up matrix

Risks	Risk Prob	Risk Impact	Risk Evolution	Risk Indicators	Mitigation Plan
Non-respect of Planning	⇒	↑	↘	# weeks of delay	<ul style="list-style-type: none"> <li>• Block 2 has been rollout 1 week in advance.</li> <li>• Planning for Block 3 is strictly followed up till now.</li> </ul>
Availability and stability of the technical infrastructure (HIS, DMZ, VPN, SIROCO architecture,...)	⇒	↑	↘	% availability	<ul style="list-style-type: none"> <li>• Connectivity checklist in place</li> <li>• Incident escalation procedure in place.</li> </ul>
Code Acceptance	⇒	↑	↘	# findings in code review	<ul style="list-style-type: none"> <li>• B1 and B2 have passed the code reviews.</li> <li>• The validated code review check list contains around 100 points</li> </ul>
BCEE/HP resources availability	↓	↑	↘		<ul style="list-style-type: none"> <li>• Daily conference calls with the offshore team has improved the communication</li> </ul>
Change Requests Management	↓	↑	↘	# Mandays	<ul style="list-style-type: none"> <li>•The estimation and validation procedure for the Change Request management in place</li> </ul>