DESJARDINS *NEXT* DELIVERY APPROACH

New Enterprise in Expansion and Transformation (NeXT)
Case Study – February 22, 2018

IMPORTANT THINGS TO KNOW

- This case study is presented by Levio, a DAC Bronze Partner, describing work that they did with Desjardins, a large Canadian financial institution.

- One of the things that Desjardins decided to do on the NeXT program was to inspire the delivery approach from of Disciplined Agile Delivery (DAD) to better fit in with their overall process efforts.

- Desjardins also decided to adopt DAD terminology (i.e Iteration, Daily Coordination Meeting) over Scrum-based terminology (i.e. Sprint and Scrum meeting). This is a common decision that has the potential that it will move teams beyond Scrum's limitations.

- Considering that the Desjardins NeXT Program runs in several deliveries, the impact and the cohabitation with the legacy systems, Desjardins reworked DAD' Inception phase to be a two-phase effort closer to the traditional Requirements and Architecture phases. This is an improvement over their previous traditional approach but is different than DA point of view.

- Nonetheless, for the new platform that they will build with their partner EIS Group, the delivery approach envisage an Initial Vision Architecture that will be proven in the early Construction phase iterations.

- This deck, without this slide, is used internally to overview NeXT.

TACTICAL SCALING FACTORS FACED BY DESJARDINS NEXT PROGRAM
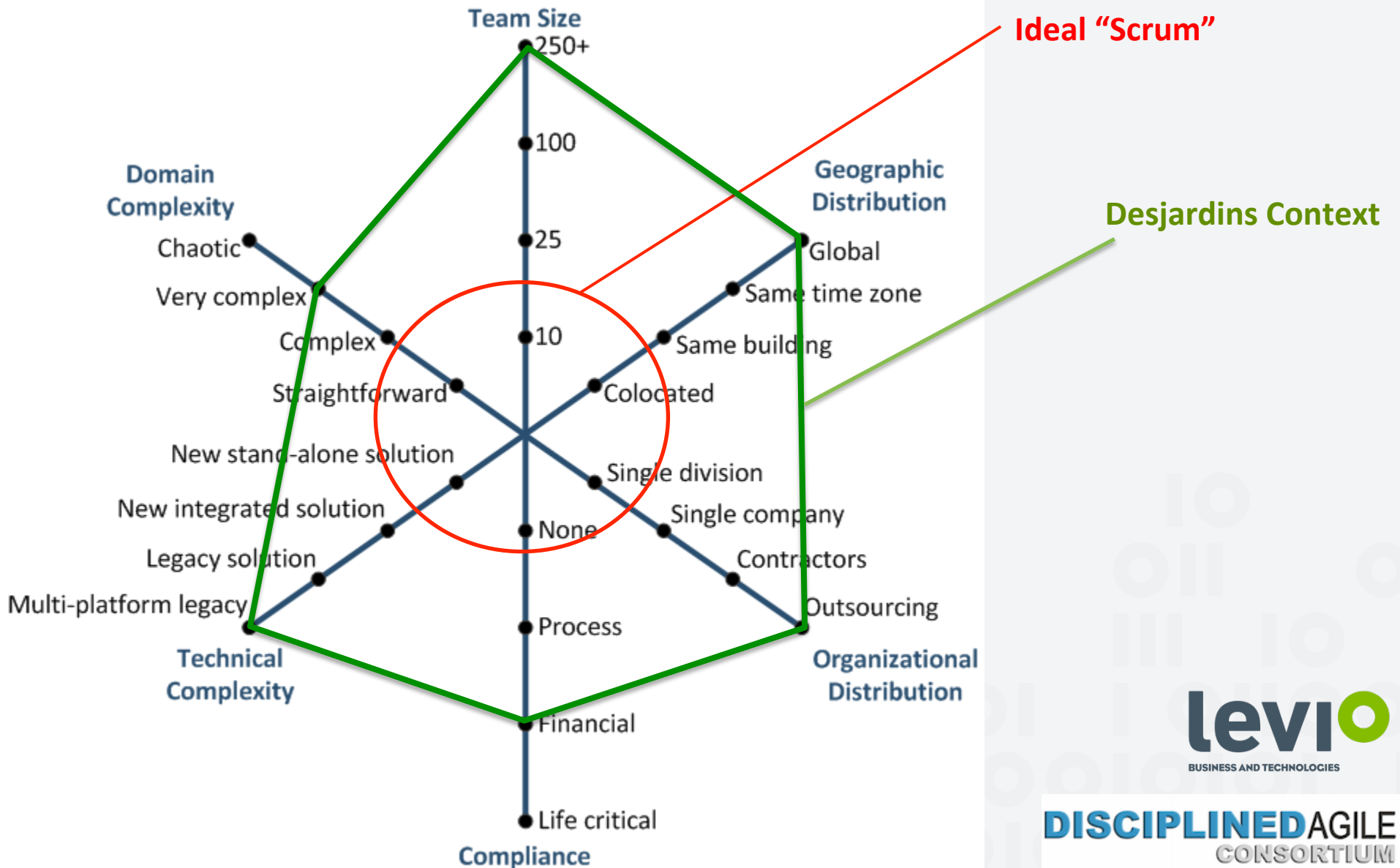
NeXT delivery approach
Orientations related to NeXT delivery approach

February 14, 2018

Based on version 1.0.1

Executive summary

A major program like NeXT has to find its own work recipe that does not include any improvisation:

- NeXT uses a pragmatic delivery approach (iterative-incremental based on Disciplined Agile Delivery) which enable establishing a work plan early, implement a rigorous follow-up of the plan and encourage mobilization and commitment of the teams to reach the delivery's objectives.

- The delivery approach is scalable and appropriate to the major program NeXT. The approach will not necessarily be the one used in DGIG 2.0.

- The moving from design to realization is a structuring step for the rest of the delivery. That moving makes it possible to have a plan and strategies for realization and testing in order to execute and monitor it.

- The implementation team is supported by QA experts and responsible for executing tests.

- A short and clear chain of governance is defined to ensure efficient communications and the application, as accurately as possible, of the guidelines.

Summary

1. **Objectives of the document**

2. **Context and guiding principles**

3. **NeXT delivery approach**

4. **Tools used by the delivery team**

5. **Governance and team structure**

6. **Next steps and conclusion**

Objectives of the document

1 Communicate the targeted approach of work related to a delivery in NeXT.

2 Standardize the understanding of every stakeholder.

3 Clarify the structure of governance of the deliveries.

Context and guiding principles
A major Program like NeXT has to find its own work recipe to reach its cruising speed.

A context of implementation constrained with limited options.

- Renewal of current insurance systems
- 450 M$ of investments

- Fixed budget, scope and deadlines
- Involvement of a new partner (EIS Group)

An approach based on 4 steps and guiding principles
to define the foundations of NeXT recipe
and the key success factors of the Program

NeXT Program **global architecture** work	Architecture detailed design of the delivery including business, solution and software components*	**Iterative** and **incremental** build of the solution, using Agile principles	**Implementation, activation and post-implementation** as the last phase of a Delivery
Business needs and initial requirements, Potential solution analysis, **Solution selection,** **Scope definition.**	*** Initial Vision Architecture for EIS Genesis Delivery** **Plans and estimates,** **Realization and testing strategies,** **Scope broken down into blocks** (delivery increments), **Initial delivery backlog set up.**	**** Proven Architecture for EIS Genesis Delivery** **Small and multidisciplinary teams** **Committed and accountable teams,** **Testing as soon as possible in the process,** **Regular code integration.**	**Risk management strategies,** **Progressive implementation,** **Progressive activation,** **Change management.**

Based on these guiding principles, NeXT uses a hybrid and pragmatic delivery approach which enable establishing a work plan early, implement a rigorous follow-up of the plan and encourage mobilization and commitment of the teams to reach the delivery's objectives

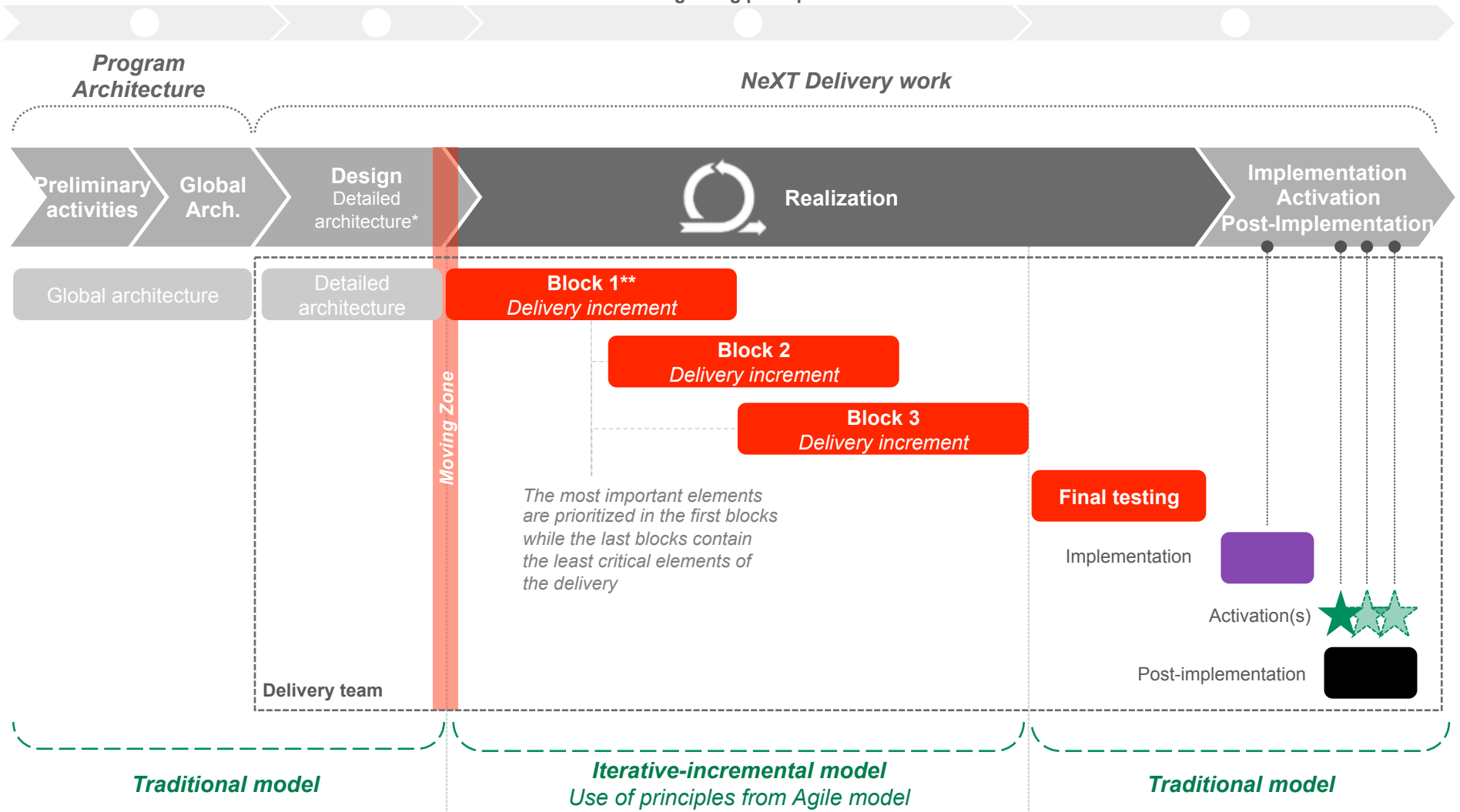# NeXT delivery approach

During the realization phase, the team focuses on achieving delivery increments.

4 guiding principles

Program Architecture

NeXT Delivery work

| Preliminary activities | Global Arch. | **Design** Detailed architecture* | Realization | Implementation Activation Post-Implementation |

Global architecture

Detailed architecture

Moving Zone

Block 1**
Delivery increment

Block 2
Delivery increment

Block 3
Delivery increment

The most important elements are prioritized in the first blocks while the last blocks contain the least critical elements of the delivery

Final testing

Implementation

Activation(s)

Post-implementation

Delivery team

Traditional model

Iterative-incremental model
Use of principles from Agile model

Traditional model

9

* Initial Vision Architecture for EIS Genesis Delivery

** The number of blocks may vary depending on the delivery context

NeXT delivery approach
Moving from design to realization is a structuring pivot point for the rest of the delivery.

Deliverables from detailed architecture

At the end of the design, **the complete solution** (identification of processes, activities and components), **the strategies** (including those of realization and testing) and **the evaluations** are documented, which allows:

Delivery Core team

→ **Prioritization** of the components,

→ Identification of the **dependencies,**

→ **Break down into blocks** (delivery increments),

→ Definition of the **milestones of the blocks,**

→ **Planning** of work,

→ **Organization of source code** and **branch management** within a delivery,

→ Identification of **integrated testing periods.**

Initial product backlog creation

Planning creation

- **Activities planning**
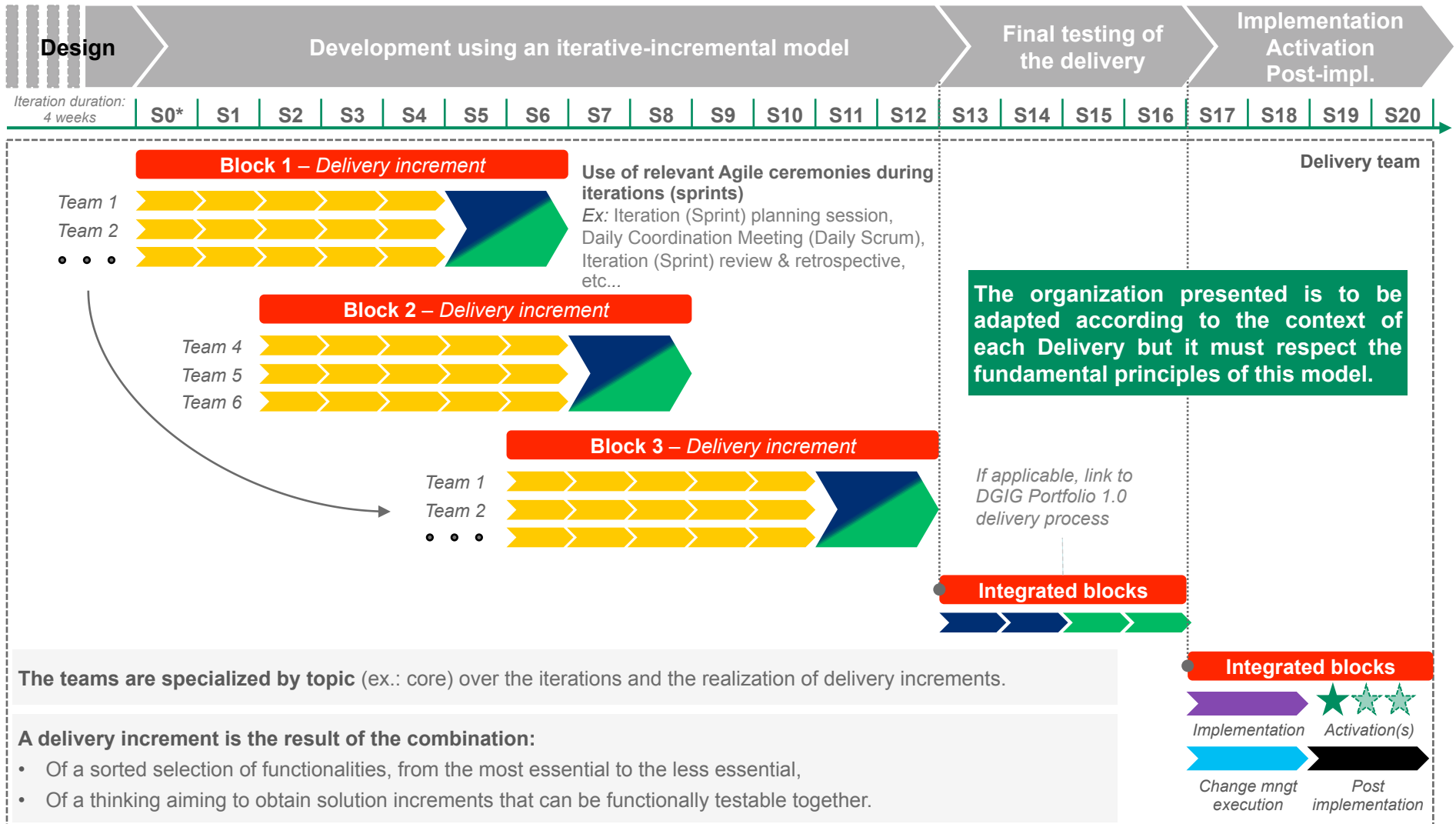- **Resource and capacity plan**
- **Assignments**

In a nutshell, this makes it possible to have a plan, execute and follow-up on it.

NeXT delivery approach

In realization, each block consists of sprints whose objective is to achieve a portion of the delivery increment.

Design	Development using an iterative-incremental model		Final testing of the delivery	Implementation Activation Post-impl.

Iteration duration: 4 weeks

S0*	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20

Delivery team

Block 1 – *Delivery increment*

Team 1
Team 2
• • •

Use of relevant Agile ceremonies during iterations (sprints)
Ex: Iteration (Sprint) planning session, Daily Coordination Meeting (Daily Scrum), Iteration (Sprint) review & retrospective, etc...

Block 2 – *Delivery increment*

Team 4
Team 5
Team 6

The organization presented is to be adapted according to the context of each Delivery but it must respect the fundamental principles of this model.

Block 3 – *Delivery increment*

Team 1
Team 2
• • •

If applicable, link to DGIG Portfolio 1.0 delivery process

Integrated blocks

Integrated blocks

Implementation   ★★★ *Activation(s)*

The teams are specialized by topic (ex.: core) over the iterations and the realization of delivery increments.

A delivery increment is the result of the combination:
- Of a sorted selection of functionalities, from the most essential to the less essential,
- Of a thinking aiming to obtain solution increments that can be functionally testable together.

Change mngt execution   *Post implementation*

NeXT delivery approach
Operational steps to deliver at a steady pace and continually monitor the trajectory against the target.

During development

Build Iteration (sprint):
- **If necessary,** the development of a component can be preceded by a dedicated sprint of analysis.
- Includes functional analysis, development, unit tests, functional tests and embedded intra-team tests of components planned and delivered using an iterative-incremental model.
- Includes continuous code integration. The level of quality is guaranteed by inter-system regression tests rolled at each sprint.
- Includes, if the context is appropriate and with an objective of an early implementation, integrated inter-team tests focused on the validation of integration points (connectivity) or common modules (non-exhaustive tests). They rely on the test strategy delivered during the design phase.
- Business people participate in the multidisciplinary team during to the design and the execution of test cases.

Integrated tests* of the delivery increment with the involvement of the users and the analysts.

Performance tests* of the delivery increment.

User Acceptance Tests* validating the completion of a block. They are short because of the early involvement of the team's business people in the delivery process and during testing.

During integrated tests and final UAT of the delivery

Final integrated tests* of the delivery increments with the involvement of the users and the analysts. They end before the start of the final UAT tests.

Final performance tests* of the delivery increments.

Final User Acceptance Tests* of the delivery increments validating the end of the realization phase. They are short because of the prior implementation of the different test cycles on each delivery increment.
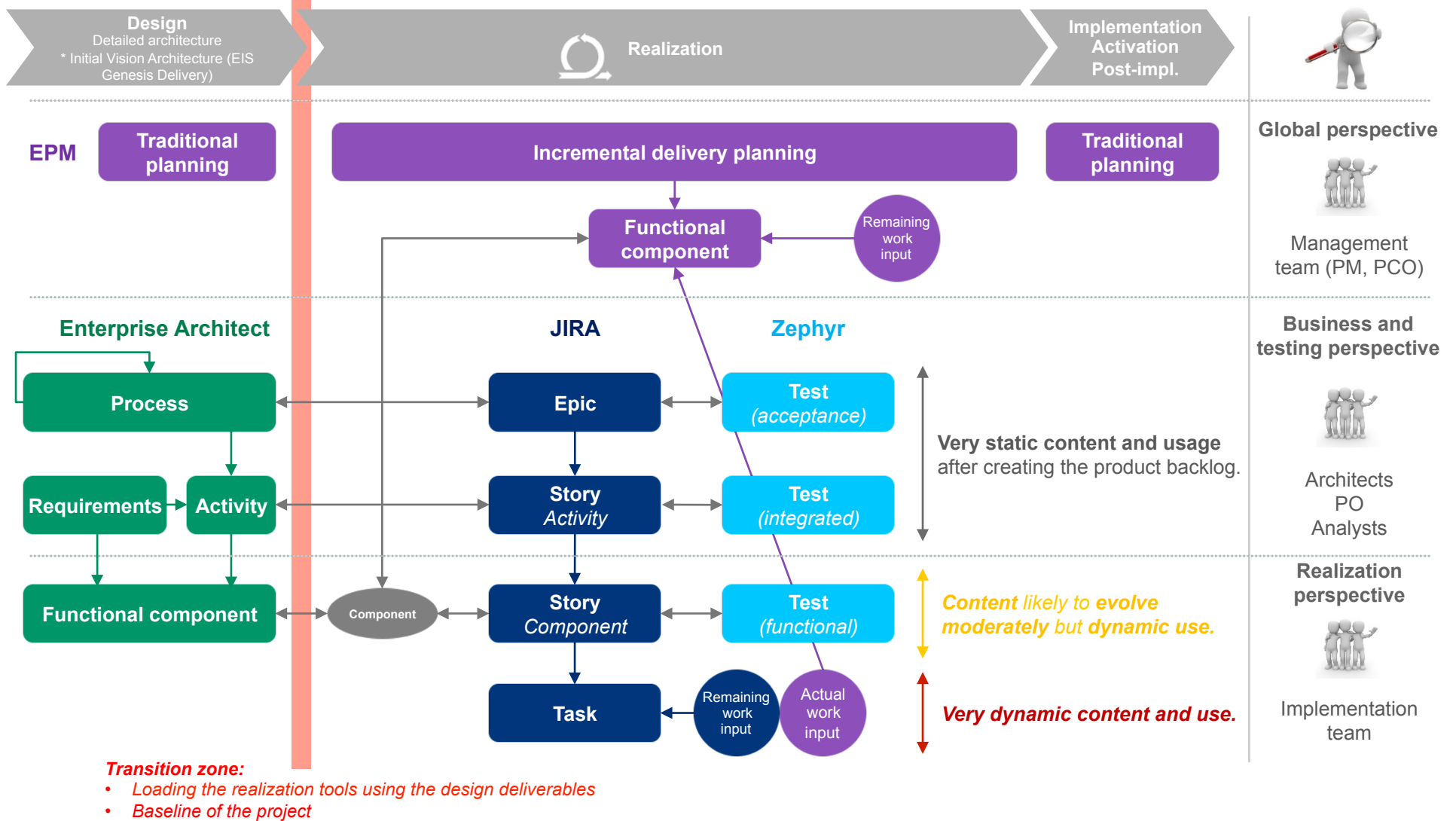
During implementation and post-implementation

Implementation, change management execution, activation(s), and **post-implementation activities** implemented by a traditional project model. These activities needs the approval of the business sectors at the end of the UAT tests.
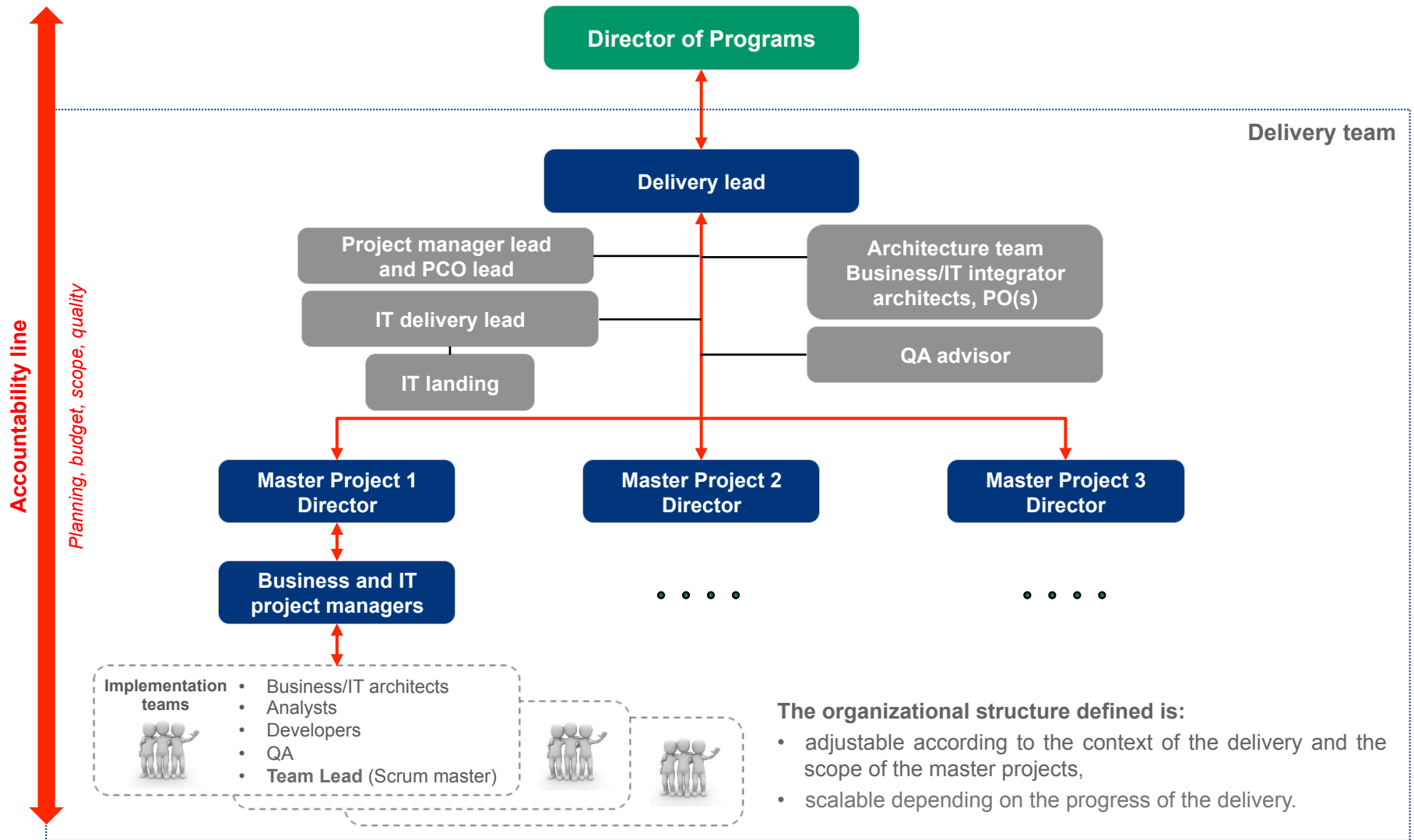
12

* Non-regression testing should executed when appropriate in the different phases of the project

Tools used by the delivery team

A simple inter-tool articulation to capitalize on the work done and to be effective on the one to do.

Design	Realization	Implementation Activation Post-impl.
Detailed architecture * Initial Vision Architecture (EIS Genesis Delivery)		

Global perspective

EPM — Traditional planning — Incremental delivery planning — Traditional planning

Functional component ← Remaining work input

Management team (PM, PCO)

Business and testing perspective

Enterprise Architect — **JIRA** — **Zephyr**

Process	Epic	Test (acceptance)
Requirements → Activity	Story *Activity*	Test (integrated)
Functional component — Component	Story *Component*	Test (functional)
	Task	

Very static content and usage after creating the product backlog.

Architects PO Analysts

Realization perspective

Content likely to evolve moderately but dynamic use.

Remaining work input — Actual work input

Very dynamic content and use.

Implementation team

Transition zone:
- *Loading the realization tools using the design deliverables*
- *Baseline of the project*

A more granular view of the interactions between the tools will be included in the appendix of the presentation

Governance and team structure

A short and clear chain of governance is defined to ensure efficient communications and the application, as accurately as possible, of the guidelines.

Director of Programs

Delivery team

Delivery lead

Project manager lead and PCO lead

Architecture team Business/IT integrator architects, PO(s)

IT delivery lead

QA advisor

IT landing

Accountability line

Planning, budget, scope, quality

Master Project 1 Director

Master Project 2 Director

Master Project 3 Director

Business and IT project managers

· · · ·   · · · ·

Implementation teams
- Business/IT architects
- Analysts
- Developers
- QA
- **Team Lead** (Scrum master)

The organizational structure defined is:
- adjustable according to the context of the delivery and the scope of the master projects,
- scalable depending on the progress of the delivery.

Next steps and conclusion

The priorities are to operationalize the approach and evaluate the adjustments to be implemented in the current deliveries.

1 Operationalize and operate the targeted approach within **Customer and Perception** delivery.

2 Identify the gaps, evaluate and validate the winning adjustments to implement for **Commercial Quebec** delivery.

Identify the gaps, evaluate and validate the winning adjustments to implement for **New Rating Solution** delivery.

* Subsequent deliveries will use the new delivery approach