

IBM Sales Cloud: Transforming the way we sell globally

DR. ASHAY SAXENA, International Business Machines Corporation

This IBM experience report focuses on providing insights from one of the largest Salesforce CRM transformations executed in the world. The objective of this program is to simplify the sales processes and increase IBM's seller time in front of clients. From an execution standpoint, the program required Salesforce platform customization, architecture design, environment setup, data integration, and frequent tooling effort following an agile mindset. All of this, including the people ecosystem embracing work-from-home, had to come together to handle an aggressive timeline for global rollout. This report deep dives into uncovering elements from a program management framework adopted precisely for such fast-track delivery.

1. INTRODUCTION

The sales, sales management, and business partner community at IBM is a diverse ~70K+ people team operating out of various geographies. Customer Relationship Management (CRM) is a key tool for this group catering to the following requirements: Lead Management, Opportunity Management, Account and Contact data from respective trusted sources, Territory Management, generating accurate pipeline as a lead-up to forecasting management and reporting management.

At the heart of every CRM, as the name implicitly suggests, it is about maintaining positive customer relationships. Salesforce is one of the most popular CRM systems since it is packed with features, such as workflow creation, collaboration tools, analytics, and a mobile-ready dashboard. We leveraged all these exciting features in our design but firmly kept the focus on user-centricity. IBM Sellers are diverse as well as heterogeneous and are spread across the globe. Catering to their needs required us to customize the solution.

To manage this development effort, we relied on a program management framework involving six transformation levers, which involved a mix of social as well as technical elements. From a scaling agile perspective, we choose to leverage Scrum of Scrums rather than any well-known prescriptive frameworks. The focus was firmly on driving agile in a bottom-up manner, where the teams lead the process and leadership lends required support, providing technical guidance on matters.

2. BACKGROUND

IBM followed legacy CRM software before 2010. At the turn of the decade, we migrated to an on-premise CRM, followed by one on the cloud in 2018, which had dedicated instances for protected groups as well. In 2019-20, the CRM Platforms team at the Chief Information Office (CIO) took up key initiatives of onboarding all sellers and sales managers from the Business Units to CRM on cloud. Around mid-year 2020, IBM took a strategic decision to launch Salesforce as its CRM of choice, christened IBM Sales Cloud (ISC) with the following objectives.

- Simplify Sales Process
- Increase seller time in front of the client
- One-CRM (consolidate 160+ sales tools)
- Live on the system – Reports & Dashboards with actionable insights for Sales teams

The program was given a challenging target to complete the transformation in one year. In this experience report, I will share details of the framework leveraged by the CRM Platforms team to make IBM Sales Cloud program successful and achieve IBM's strategic objectives.

The transformation team involved representation from program management, business product ownership and development leadership. A group of architects provided overall technical guidance and direction in the setup. IBM, being an agile organization, has significant agile process maturity which enables teams to lead the process themselves.

3. TRANSFORMATION LEVERS

As a Business Transformation Consultant, I was driving the transformation effort alongside Program Managers. Given our rich prior consulting and coaching experience, we leveraged a transformation-levers-based framework. We were keen to ensure that the framework covered all aspects of agile values and principles and not just introduce some practices in haste. As per McKinsey’s [article](#), “A comprehensive transformation touches every facet of the organization, including people, process, strategy, structure, and technology” (Brosseau et al. 2019).

The framework allowed us to take a holistic perspective, focusing on both technical and social elements. IBM Sales Cloud (ISC) transformation levers (refer Figure 1) are broadly structured along six intertwined themes: (1) Team setup, (2) Business Demand & Process, (3) Architecture, (4) Engineering Practices & Tooling, (5) Governance and (6) Culture.



Figure 1. IBM Sales Cloud Transformation levers-based Program Management Framework

3.1 Team setup

ISC program team involves multiple stakeholders with a clear definition of roles and responsibilities to help streamline the program. Figure 2 provides a high-level overview of the team structure and inter-connectedness across verticals.

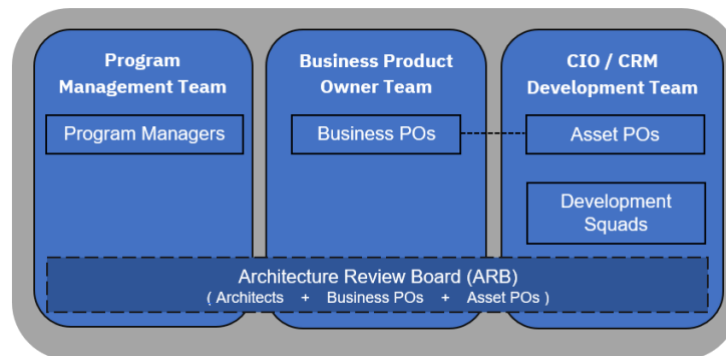


Figure 2. IBM Sales Cloud Program Team

An integral aspect of the structure is the CIO/CRM development teams, which form the crux for delivering the business functionalities for the ISC program. The overall setup has 8 development teams to work on CRM functionality, 6 in India and 2 in USA. There are other groups working on Partner relationship module (Salesforce PRM), Tableau CRM, and User Enablement besides these 8 development teams.

- Followed workstream-based teams (refer Figure 3) with full-stack development skills: provides end-to-end ownership for a specific CRM module, such as Accounts, Contacts & Leads, Opportunities, and Territories. Existing CRM domain business knowledge helped our development teams significantly.
- A key principle around team set-up was ‘independent work-streams’ so we could minimize cross-team dependencies. It also helped in moving shippable increments to higher environments faster. Rather than following component team setup where there is a dedicated UI team, Middleware team and a separate DevOps team, the focus was to reduce this extensive dependency management and let each team have all the skills required to deliver increment(s) every sprint.
- Experienced team members for a given domain from legacy CRM setup were combined with additional new members to form cross-functional development teams. This enabled leadership to come from within teams, especially when it comes to following agile as a process.

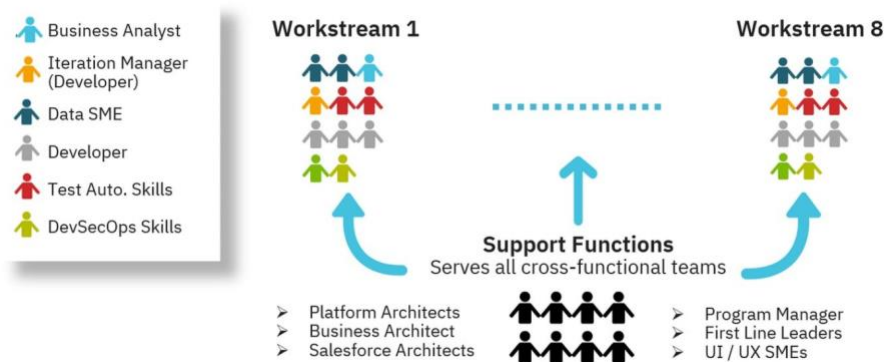


Figure 3. IBM Sales Cloud Development Teams

3.2 Business Demand & Process

As the teams ramped up on Salesforce knowledge, we started laying the groundwork to create a product backlog and setting up an iterative process for the program to follow (refer Figure 4). A high-level plan in areas of Architecture, Staff enablement, Development Environment and User experience was put in place.

- The **30-60-90 days backlog**, which focused on initial setups, became our first release plan.
- Followed a third-party Agile application lifecycle management (ALM) tool for the transformation journey from the beginning:
 - Structure followed for creating a funnel of work items: Initiative -> Epics -> Stories
 - Initiatives have been owned by Business POs
 - Epics have been owned by Asset POs
 - Since the inception of development work, initiatives have been created by Business PO, and Asset PO is taking it forward by defining epic(s) to fulfil the requirement. Epics are elaborated in consultation with business stakeholders and go through a formal

Architecture Review Board (ARB) process. Once approved, epics are moved to ‘squad planning’ when the development team(s) dissect the epic and come up with work items in the form of stories. When the team starts to pull stories for an iteration, the epic moves to ‘in progress’ and is subsequently marked ‘done’ when all the underlying stories have been completed, and validation is done by Asset PO.

- Conducted **Program iteration planning** and **showcase** by all teams together to keep high-level visibility of the progress being made and ensure the right prioritization of future work.
- Focused on achieving ‘**Definition of Done**’ to ensure quality deliverables.
- Even though this was packaged software development, focus was setting up a process to deploy to the Production environment every iteration, paving the way for faster time-to-market and business stakeholder feedback loop.

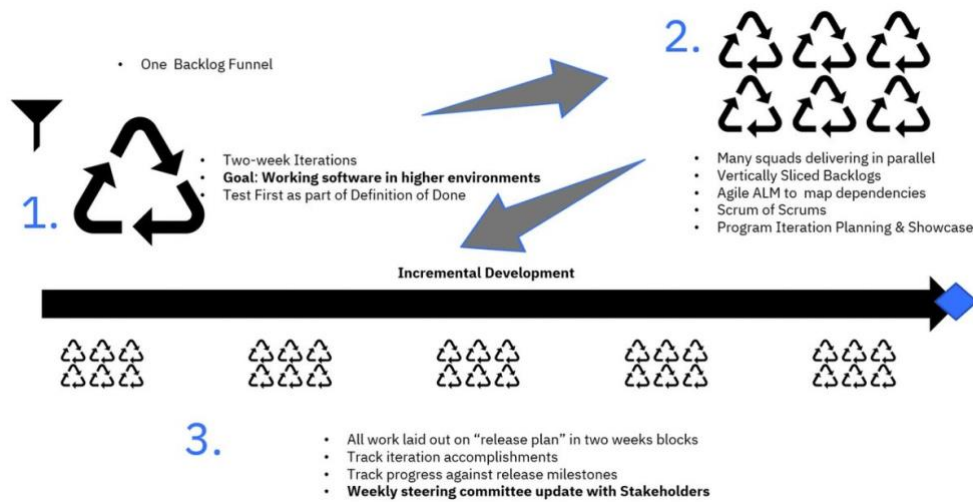


Figure 4. Agile delivery process adopted by IBM Sales Cloud teams

Which scaling agile framework did IBM leverage for this transformation?

None, IBM did not follow any defined scaled agile frameworks. Instead, the focus was firmly on following Scrum of Scrums (SoS) and extreme programming (XP) practices in everything the development team did. SoS worked out well for our setup involving 8 development teams having a high level of agile process maturity and spread across two sites.

3.3 Architecture

This theme revolves around setting up the basic infrastructure to drive the transformation effort. In this regard, we had set up a formal agile cadence and meetings to establish building blocks (including but not limited to, DevOps tooling and common development practices, data model, integration with trusted sources and crafting backlog for development teams) at the program level. This ‘release plan’ involved multiple two-week iteration blocks, which allowed us to track accomplishments and progress against release milestones.

- Organized **Illuminate sessions** led by Business Product Owners (POs), Salesforce & Platform Architects to (a) define seller personas and journey mapping to enable UI/UX interface design, and (b) set up the high-level operating model for the program.

- As an outcome of the Illuminate sessions, business architecture was presented three months into the program, which became the basis for scaling up the ISC program. Scalable Architecture and modularity principles have been the core pillars of this endeavour.
- Integration & Data Architecture was envisioned by CRM Enterprise architect with a focus on data modelling with source CRM as well as trusted sources, middleware architecture and DevSecOps practices.
- Setup of **Architecture Review Board (ARB)** where architects across functions, including Business Architect, Platform Architect, Enterprise CRM Architect, and Tooling Architect came on board (refer Figure 5). Aside from their key role in the finalization of architecture decisions, the board also enabled governance from a change management perspective at the platform level.
 - Conducted ARB meetings on alternate days of each week (M-W-F) to finalize key architectural decisions and the epic approval process for new work items being elaborated by Asset POs.
- Weekly **steering committee updates** at Vice President level enabled us to keep everyone informed regularly.

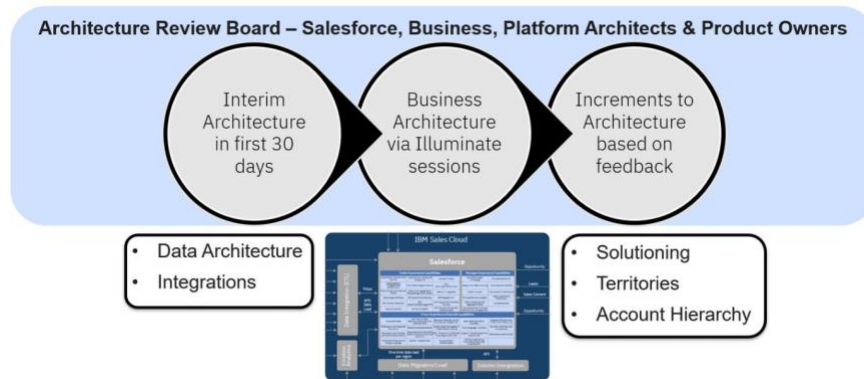


Figure 5. Key Principles for IBM Sales Cloud Architecture

3.4 Engineering Practices & Tooling

Any enterprise transformation program must have solid infrastructure as far as the development and testing environment is concerned to iteratively deliver to production rather than remaining focused on big-bang releases. Hence, we enabled the setting up of environments in the first few weeks of the program. We kicked off engineering practices and tooling setup subsequently. Setting up cross-functional development teams helped us align on key principles for engineering practices and tooling.

- **Development environments** were created with a clear strategy of code promotion and readiness from day one. While two main pre-prod environments (Integration & Staging) were set up for inter-team collaboration and testing, sandboxes were provisioned to teams on a need basis.
- The GitHub code repository was structured keeping in mind the Salesforce principle of common code at the lowest level. Development naming conventions and principles were followed right from the beginning to ensure good quality of code being committed. Salesforce offered an easy-to-use **Developer Experience (DX) tooling** that enabled developers to deploy code onto Staging and Production in a systematic manner.
- **A single sign-on setup** was put in place, which enabled individuals to seamlessly access the platform leveraging their enterprise IDs and made the overall experience secure while logging into the Salesforce platform.

- From a **compliance** standpoint, regulations such as Application Systems Control and Auditability (ASCA), IT Security Standard (ITSS) and Business Information Security Officer (BISO) were accomplished. For instance, ASCA provides an independent assessment of the controls and auditability of IBM applications. It enabled us to ensure control points for trusted source integrations with the ISC platform are secure and aligned with architectural design. Along with functional implementation, development teams handled these compliance requirements without raising any exceptions.
- **Test automation** is one of the key engineering practices that we started as early as the development kicked off.
- Salesforce mandates a minimum of **75% code coverage** to be able to deploy code onto production, which ensured our unit tests for the custom classes created covered all required scenarios.
- We employed the **microservices architecture** to set up integration between the ISC environments and trusted sources. These services are managed via a dedicated **Jenkins CI-CD pipeline** (refer Figure 6 for salient features) and were developed keeping in mind the error handling for responses from the external trusted source APIs.
- A **customized Wiki** was set up to guide development teams on best practices and to maintain order for the entire process of Development to Deployment. The wiki also clearly specified the requirements for the entire developer set-up to start contributing to the project.
- One of the most important engineering practices is to promote **code to production** as often as possible. We made sure the latest code was deployed to production once per iteration, gradually shifted to once per week and eventually incorporated an on-demand basis specific team's requirement.



Figure 6. **Benefits of Middleware CI-CD pipeline for IBM Sales Cloud**

3.5 Governance

We completely relied on an Agile ALM tool as a single source of truth to ensure all the stakeholders have a unified view of the progress being made on the program. It provided us with a visual way of looking at key dependencies and analyzing the movement of program requirements from one state to another. It enabled ease of sharing progress-related information with key stakeholders (refer Figure 7) and crafting a burn-up chart for the program.

- Agile ALM dashboards provided a one-stop reference point for all the stakeholders to gauge program-level progress during daily Scrum of Scrums discussions.
- Agile ALM helped simplify dependency management across stakeholders in key functional areas.

- We followed vertical slicing for breaking down work to ensure minimal dependency across teams. Several practices, such as weekly global cross-squad calls, daily Scrum of Scrums and a dedicated messaging channel helped us manage dependent work items.
- Leveraged Agile ALM as a program management tool such that initiatives and epics dashboard provided a visual way of analyzing the movement of program requirements from one state to another.
- Followed backlog funnel tracking and flow, which meant having a clear view of epics across to-do/discovery/squad planning stages at a development team level.
- Conducted weekly program review meetings with key business stakeholders to assess progress at the program level.

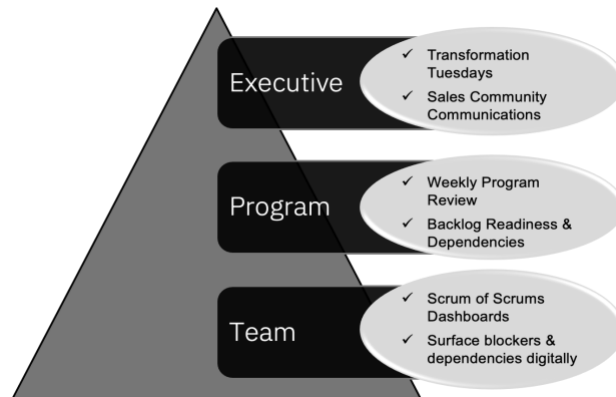


Figure 7. Governance for IBM Sales Cloud Program

What are the key signals that a company is getting 'bottom-up' agile right?

From an IBM perspective, teams come with significant agile process maturity. This allows us to build leadership within teams for practicing agile in the true spirit. Leadership support is key and their focus is more on resolving technical concerns/blockers rather than dictating the process.

3.6 Culture

Culture is an integral aspect of any transformation journey. In our view, culture is what individuals experience daily in their interactions with other team members, peers, leaders and stakeholders. Hence, we took a micro perspective of this multi-dimensional construct, where the focus has been kept on teams (refer framework from Spencer Stuart in Figure 8). We inculcated the practice of encouraging teams to bring forward their issues so we can provide solutions to them, going beyond just engaging in multiple discussions.

- Teams were encouraged to be transparent about the issues being faced in ramping up on the Salesforce platform and/or understanding the business requirements. Any issues that surfaced were addressed upfront.
- During our quarterly cross-squad retrospective, we encouraged team members to raise the ongoing issues so we could address any impending challenges being faced by them.
- First Line Leaders and Technical Program Manager were hands-on in helping teams when faced with technical issues.

- Iteration Managers for each team were individuals with good technical experience and ensured having a great outlook on how teams were progressing during their ramp-up and afterwards. This helped us to establish good team practices from the ground up and scale them eventually.
- Focused on keeping the core principle of agility at the forefront: teams have been encouraged to experiment, learn from it, and make suitable changes (if needed) in an iterative manner while delivering complicated functionalities for their workstreams.
- We encouraged a high-performance culture within teams. The teams were a good mix of members with prior CRM domain experience and new developers with full-stack development experience. It helped each team mature into a highly motivated, independent unit with the ethos of one-team behaviour in every action.

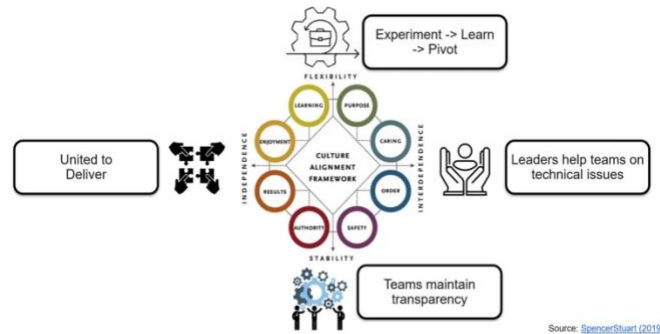


Figure 8. One-Team Culture for IBM Sales Cloud Program

4. WHAT WE LEARNED

Although strong transformation levers have been leveraged, I experienced a lot of learning through this journey. Our experiences pre and post- first major release offer some insights in this regard.

Pre-release

- Although the Illuminate sessions outcome provided us with insights on user personas and high-level operating model for the program, finalization of business architecture remained a key concern. The view provided was at a very high level and was not as helpful as expected in backlog elaboration and solutioning. The learning for us was that business architecture also needs to be elaborated in an agile manner through incremental updates, modular design, and scalable implementation.
- ISC was set a fixed schedule of one year from the date of launch to all geographies go-live. The entire program was aligned to handle this expectation, which resulted in stretch efforts to keep up with the schedule. We couldn't have met program objectives if we did not follow an iterative and incremental approach to development with working software in production every iteration.

Post first major release

- Territory load in production was designed to be done following country-level account hierarchy (as decided based on ISC architecture). However, the existing trusted source data on territories is based on city-specific accounts. This resulted in sellers seeing incorrect accounts on their territory and hence did not meet their needs. The mismatch had a cascading negative impact on data visible on dashboards. The learning for us was that business units needed to be involved more frequently in initiative/epic elaboration, solutioning and soliciting user feedback. This was significantly improved after our first market go-live.

- Another learning after our first major release centered on Business Unit (BU) specific customizations required for the platform. For instance, one of the BUs relies on specific tag values to view their territory, segment, and industry details. The learning for us was that our initial approach to going with a standardized one-size-fits-all process across the organization had to be revisited. We initially thought a standardized approach would ensure minimal customization and further simplification of the sales process. However, some of the BUs needed more time to align to one process, and it resulted in tactical solutions being implemented by customizing them for their requirements. Although it took us away from our initial approach of minimal customization, it was also learning to keep exploring options to support all BUs while maintaining a simplified process implementation.
- Salesforce had some product limitations. The tool that we leverage for deployment (DX) does not support all metadata, and hence deployments to higher environments required a few manual steps. Despite the limitation, we worked towards deploying at least once a week to production. Then, there were some limitations related to data load volumes and time taken due to process/flow implementation. We identified dedicated slots with minimal seller activity to work around some of these limitations.

5. BUSINESS OUTCOMES

Relying on transformation levers, we have been able to deliver a success story since our very first major release. As of today, approximately 30K sellers from all Business Units are living on the platform as envisioned. The program also involved data migration from the current CRM system(s) to ISC for each geography. 10M+ records across all objects are migrated and streamed in real-time to downstream applications.

The Test-first approach helped in ensuring the quality of delivery was maintained throughout the program. Development teams received only one Severity 1 incident related to features developed. At the same time, all work passed the scrutiny of GEO-specific data visibility and other compliance requirements. To top it up, seller feedback has been extremely positive and suggests having a great live-in experience on the platform.

6. ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my teams, colleagues, peers and partners at IBM - the prime movers behind this transformation journey. A huge shout-out to my Product Owners group - Kathleen, Poovannan, Mike, Glenn, Bridget, Walt, Ram and others - who have been instrumental in defining the platform experience. Special thanks to Jayesh Kadam for the insightful discussions and candid feedback on the report. Finally, I would like to acknowledge my shepherd, Sibylle Peter, for her keen insights, questions, and edits that made a significant difference in this experience report. We couldn't have done it without you!

REFERENCES

- Brosseau, D., Ebrahim, S., Handscomb, C., & Thaker, S. (2019). The journey to an agile organization. *McKinsey & Company, May, 10*, 14-27.
- Spencer Stuart (2019). Culture Alignment Framework. <https://www.spencerstuart.com/what-we-do/our-capabilities/leadership-consulting/organizational-culture>